– EPE – NEW NAME NEW DESIGN!

PE Practical Electronics

The UK's premier electronics and computing maker magazine

PIC n' Mix
Connecting I²C LCD displays

Electronic Building Blocks
Fun with LEDs

Circuit Surgery
Differential amplifiers

Micromite
Fonts, files and temperature

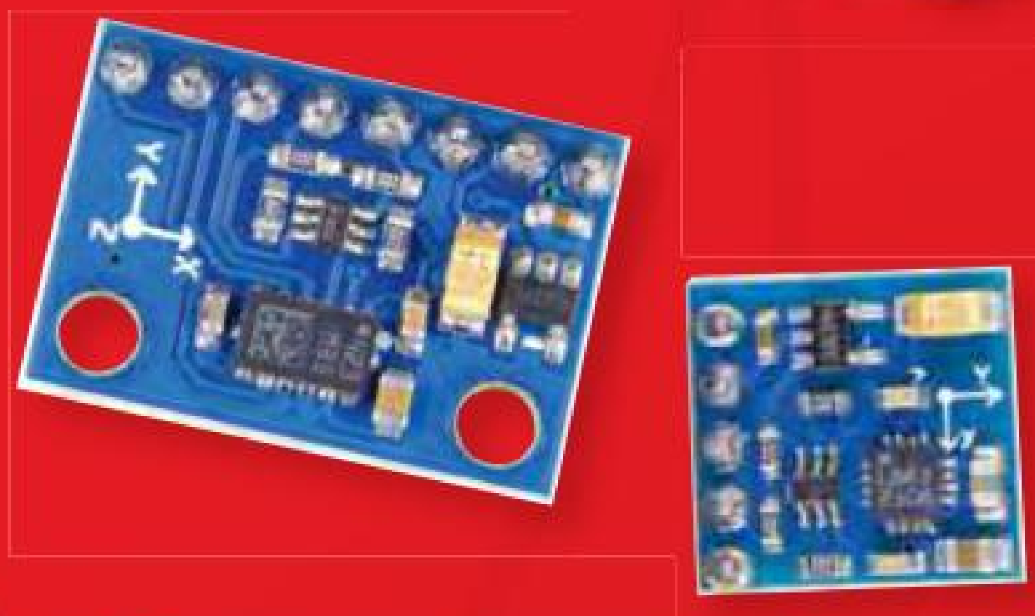
Hello from PE
Temp: 21.6°C
Grove-LCD RGB Backlight

Choosing and identifying stepper motors

SILICON CHIP
TINNITUS & INSOMNIA KILLER!
White
Pink

Tinnitus & Insomnia Killer

WIN!
Microchip SAM R30M Xplained Pro Evaluation Kit
WIN!



Electronic compasses

MAXIMITE
COLOUR MAXIMITE

Build your own retro Colour Maximite Computer!

4 433 619Hz
MHz
kHz
OK
CANCEL
GPS-SYNCHED PROGRAMMABLE FREQUENCY REFERENCE
SILICON CHIP

GPS-synced Frequency Reference

PLUS!

Net Work – Surveillance tech
Techno Talk – VT100 Emulator
Audio Out – Speaker building

WIN!
$50 of PCB prototyping from PCBWay

Nov 2019 £4.99
11
9 772632 573016

With Over 250 Million Units in Stock, We've Got You Covered

Delivering You the World's Largest Inventory of Microchip Products

www.microchip.com/learn-more

The Microchip name and logo and the Microchip logo are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.
All other trademarks are the property of their registered owners.
© 2018 Microchip Technology Inc. All rights reserved. DS00002552A. MEC2226Eng10/18

Practical Electronics

Volume 48. No. 11
November 2019
ISSN 2632 573X

Contents

SILICON CHIP
TINNITUS & INSOMNIA KILLER!
White
Pink
MAXIMITE
COLOUR MAXIMITE
4 433 619Hz
MHz
kHz
OK
CANCEL
GPS-SYNCHED PROGRAMMABLE FREQUENCY REFERENCE
TOUCH SCREEN FOR SETTINGS

Projects and Circuits

Programmable GPS-synced Frequency Reference – Part 2 by Tim Blythman 14
Detailed assembly instructions and further information on how to use this super instrument, including all the various customisable settings.

Using Cheap Asian Electronic Modules – Part 20 by Jim Rowe 22
Learn to use Elecrow's GY-271 and GY-511 low-cost electronic compass modules.

Tinnitus & Insomnia Killer by John Clarke 28
Suffer from Tinnitus or insomnia? This device produces either pink or white noise, which may help you get some relief.

Colour Maximite Computer – Part 1 by Phil Boyce 36
This project has fun written all over it! Build your own standalone computer based on a powerful PIC32 microcontroller running the easy-to-use MMBASIC language.

Series, Features and Columns

Techno Talk by Mark Nelson 10
Greater flexibility

Net Work by Alan Winstanley 12
The extraordinary ambition of surveillance technology

Using Stepper Motors by Paul Cooper 42
Choosing and identifying stepper motors

Circuit Surgery by Ian Bell 46
Strain gauges and differential signals

PIC n' MIX by Mike Hibbett 50
Small, cheap and powerful – Part 4

Audio Out by Jake Rothman 54
Speaker nuts and bolts – Part 1

Make it with Micromite by Phil Boyce 60
Part 10: Adding fonts, measuring temperature, saving files

Max's Cool Beans by Max The Magnificent 66
The real work begins!

Electronic Building Blocks by Julian Edgar 76
Four simple and cheap LED kits

Regulars and Services

Exclusive PCBWay reader offer 3
Win one of six $50 credits to use with PCBWay's superb PCB prototyping service

Subscribe to Practical Electronics and save money 4

Reader services – Editorial and Advertising Departments 7

Editorial 7
Downloading files

Practical Electronics back issues CD-ROM – great 15-year deal! 11

Exclusive Microchip reader offer 21
Win a Microchip SAM R30M Xplained Pro Evaluation Kit

PE Teach-In 8 27

Practical Electronics – get your back issues here! 59

Teach-In bundle – what a bargain! 65

Practical Electronics CD-ROMS for electronics 70
A superb range of CD-ROMs for hobbyists, students and engineers

Direct Book Service 73
Build your library of carefully chosen technical books

Practical Electronics PCB Service 78
PCBs for Practical Electronics projects

Classified ads and Advertiser index 79

Next month! – highlights of our next issue of Practical Electronics 80

Stepper motor image on cover and contents page courtesy of Pololu Robotics & Electronics, pololu.com

© Electron Publishing Limited 2019

Copyright in all drawings, photographs, articles, technical designs, software and intellectual property published in *Practical Electronics* is fully protected, and reproduction or imitation in whole or in part are expressly forbidden.

The December 2019 issue of *Practical Electronics* will be published on Thursday, 7 November 2019 – see page 80.

Quasar Electronics Limited
PO Box 6935, Bishops Stortford
CM23 4WP, United Kingdom

Tel: 01279 467799
E-mail: sales@quasarelectronics.co.uk
Web: quasarelectronics.co.uk

All prices INCLUDE 20.0% VAT. Free UK delivery on orders over £48
Postage & Packing Options (Up to 0.5Kg gross weight): **UK Standard 3-7 Day Delivery -** £4.95; **UK Mainland Next Day Delivery -** £9.95; **Europe (EU) -** £12.95; **Rest of World -** £14.95 (up to 0.5Kg).
!! Order online for reduced price postage and fast despatch !!
Payment: **We accept all major credit/debit cards. Make cheques/PO's payable to Quasar Electronics Limited.**
Please visit our online shop now for full details of over 1000 electronic kits, projects, modules and publications. Discounts for bulk quantities.

QuasarElectronics.co.uk

Solutions for Home, Education & Industry Since 1993

Official Main Dealer stocking the full range of Kits, Modules, Robots, Instruments, tools and much, much more...

PIC & ATMEL Programmers

We have a wide range of PIC, ATMEL Arduino and Raspberry Pi projects.

PIC Programmer & Experimenter Board

Great learning tool. Includes programming examples and a reprogrammable 16F627 Flash Microcontroller. Test buttons & LED indicators. Software to compile & program your source code is included. Supply: 12-15Vdc. Pre-assembled and ready to use.
Order Code: VM111 - ~~*£38.88*~~ ***£35.94***

USB PIC Programmer and Tutor Board

The only tutorial project board you need to take your first steps into Microchip PIC programming using a PIC16F882 (included). Later you can use it for more advanced programming. Programs all the devices a Microchip PICKIT2® can! Use the free Microchip tools for PICKit2™ & MPLAB® IDE environment.
*Order Code: EDU10 - **£46.74***

USB /Serial Port PIC Programmer

Fast programming. Wide range of PICs supported (see website for details). Free Windows software & ICSP header cable. USB or Serial connection. ZIF Socket, leads, PSU not included.
Kit Order Code: 3149EKT - ~~*£49.96*~~ ***£29.95***
*Assembled Order Code: AS3149E - **£44.95***
Assembled with ZIF socket Order Code: AS3149EZIF - ~~*£74.96*~~ ***£49.95***

PICKit™2 USB PIC Programmer Module

Versatile, low cost, PICKit™2 Development Programmer. Programs all the devices a Microchip PICKIT2 programmer can. Onboard sockets & ICSP header. USB powered.
*Assembled Order Code: VM203 - **£35.94***

Controllers & Loggers

Here are just a few of the controller and data acquisition and control units we have. See website for full details. 12Vdc PSU for all units: Order Code 660.446UK £10.68

USB Experiment Interface Board

Updated Version! 5 digital inputs, 8 digital outputs plus two analogue inputs and two analogue outputs. 8 bit resolution. DLL.
Kit Order Code: K8055N - ~~*£39.95*~~ ***£22.20***
*Assembled Order Code: VM110N - **£35.94***

2-Channel High Current UHF RC Set

State-of-the-art high security. Momentary or latching relay outputs rated to switch up to 240Vac @ 12 Amps. Range up to 40m. 15 Tx's can be learnt by one Rx. Kit includes one Tx (more available separately). 9-15Vdc.
*Kit Order Code: 8157KT - **£44.95***
*Assembled Order Code: AS8157 - **£49.96***

Computer Temperature Data Logger

Serial port 4-ch temperature logger. °C/°F. Continuously log up to 4 sensors located 200m+ from board. Choice of free software applications downloads for storing/using data. PCB just 45x45mm. Powered by PC. Includes one DS18S20 sensor.
Kit Order Code: 3145KT - ~~*£19.95*~~ ***£16.97***
*Assembled Order Code: AS3145 - **£19.96***
*Additional DS18S20 Sensors - **£4.96 each***

8-Channel Ethernet Relay Card Module

Connect to your router with standard network cable. Operate the 8 relays or check the status of input from anywhere in world. Use almost any internet browser, even mobile devices. Email status reports, programmable timers... Test software & DLL online.
*Assembled Order Code: VM201 - **£130.80***

Computer Controlled / Standalone Unipolar Stepper Motor Driver

Drives any 5-35Vdc 5, 6 or 8-lead unipolar stepper motor rated up to 6 Amps. Provides speed and direction control. Operates in stand-alone or PC-controlled mode for CNC use. Connect up to six boards to a single parallel port. Board supply: 9Vdc. PCB: 80x50mm.
*Kit Order Code: 3179KT - **£15.26***
*Assembled Order Code: AS3179 - **£22.26***

Card Sales & Enquiries 01279 467799

Bidirectional DC Motor Speed Controller

Control the speed of most common DC motors (rated up to 32Vdc/5A) in both the forward and reverse directions. The range of control is from fully OFF to fully ON in both directions. The direction and speed are controlled using a single potentiometer. Screw terminal block for connections. PCB: 90x42mm.
*Kit Order Code: 3166KT - **£19.99***
*Assembled Order Code: AS3166 - **£29.99***

8-Ch Serial Port Isolated I/O Relay Module

Computer controlled 8 channel relay board. 5A mains rated relay outputs and 4 opto-isolated digital inputs (for monitoring switch states, etc). Useful in a variety of control and sensing applications. Programmed via serial port (use our free Windows interface, terminal emulator or batch files). Serial cable can be up to 35m long. Includes plastic case 130x100x30mm. Power: 12Vdc, 500mA.
*Kit Order Code: 3108KT - **£74.95***
*Assembled Order Code: AS3108 - **£89.95***

8-Channel RF Remote Control Set

Control 8 onboard relays with included RF remote control unit. Toggle or momentary mode for each output. Up to 50m range. Board Supply: 12Vac, 500mA.
*Assembled Order Code: VM118 - **£71.94***

Temperature Monitor & Relay Controller

Computer serial port temperature monitor & relay controller. Accepts up to four Dallas DS18S20 / DS18B20 digital thermometer sensors (1 included). Four relay outputs are independent of the sensors giving flexibility to setup the linkage any way you choose. Commands for reading temperature / controlling relays are simple text strings sent using a simple terminal or coms program (e.g. HyperTerminal) or our free Windows application. Supply: 12Vdc.
Kit Order Code: 3190KT - ~~*£79.96*~~ ***£47.95***
*Assembled Order Code: AS3190 - **£59.95***

3x5Amp RGB LED Controller with RS232

3 independent high power channels. Preprogrammed or user-editable light sequences. Standalone or 2-wire serial interface for microcontroller or PC communication with simple command set. Suits common anode RGB LED strips, LEDs, incandescent bulbs. 12A total max. Supply: 12Vdc. 69x56x18mm
*Kit Order Code: 8191KT - **£24.95***
*Assembled Order Code: AS8191 - **£27.95***

Soundtronics.co.uk

Build your dream modular analogue synth from our range of over 80 kits, including front panels, components and PCBs

Components for the synth builder and hobbyist
- Resistors and Capacitors
- Diodes, Transistors and ICs
- Switches and Potentiometers
- Knobs, sockets, connectors, leads and more

Order online at
www.soundtronics.co.uk
Unit 29, Webb Ellis Business Park
Rugby, Warwickshire
CV21 2NP
Tel: 01788 878 002

Soundtronics MIDI Ultimate Analogue Synth

As featured in Practical Electronics February – July 2019. Available as PCB, components, case or full kits

- Three VCOs
- Mixer
- 12dB VCF
- Two LFOs
- VCA
- Two ADSRs
- Sample & Hold
- Noise Source
- Repeat Gate Generator
- Echo Effect
- MIDI Interface
- Headphone Amplifier
- Toggle Switch Patching

PCBWay

Win $50 worth of PCBWay prototyping!

PCBWay is offering Practical Electronics readers the chance to win one of six $50 credits to use with their world-leading PCB prototyping service.

Based in Shenzhen, China, PCBWay not only manufactures and assembles PCBs, but also offers its clients the most user-friendly, web-based interactive design system across the industry.

Online Quote — Upload PCB File — Order Review — Payment — Fabrication — Delivery — Confirm Received

PCBWay's online web system lets you instantly get a quote for your PCB or PCBA order, and then track the online the status of its fabrication – all via your personal PCBWay account panel. After your PCBs have been dispatched, their shipping status can be checked on your order management page.

Our online facilities help keep everything under control, saving you time and energy all too often wasted in a frustrating loop of emails and phone calls.

How to enter

For your chance to win a $50 free trial of PCBWay's prototyping service, just create your own PCBWay account at:

http://bit.ly/pe-nov19-pcbway

Closing date: The closing date for this offer is 1 November 2019

– EPE – NEW NAME NEW DESIGN!

Practical Electronics

UK readers SAVE £1 on every issue

Take out a one-year subscription and save £10 over the year.
Even better – save £1 per issue if you subscribe for two years – a total saving of £24.

Overseas rates represent exceptional value

You also:
- **Avoid any cover price increase for the duration of your subscription**
- **Get your magazine delivered to your door each month**
- **Guarantee a copy, even if the newsagents sell out**

Order by:
- **Phone or post with a cheque, postal order or credit card**
- **Or order online at: www.epemag.com (go to our secure server-backed online shop)**

Practical Electronics

SUBSCRIPTION PRICES

Subscriptions for delivery direct to any address in:

UK: 6-months £26.99, 12-months £49.85, 24-months £94.99

Europe Airmail: 6-months £30.99, 12-months £57.99, 24-months £109.99

Rest Of The World Airmail: 6-months £37.99, 12-months £70.99, 24-months £135.99

Cheques (in **£ sterling only**) payable to *Practical Electronics* and sent to: Wimborne Publishing Ltd, 113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU.

Tel: 01202 880299. **Fax:** 01202 843233.
Email: stewart.kearn@wimborne.co.uk.

Also via our secure online shop at**: www.epemag.com**.

Subscriptions start with the next available issue. We accept MasterCard or Visa.
(For past issues see the Back Issues page.)

ONLINE SUBSCRIPTIONS

Electronic subscriptions (PDF download or online) are available from £20.99 for 12 months, more details at: **www.epemag.com**

SUBSCRIPTION ORDER FORM

☐ **6 Months: UK £26.99, Europe £30.99 (Airmail), Rest of the World £37.99 (Airmail)**
☐ **1 Year: UK £49.85, Europe £57.99 (Airmail), Rest of the World £70.99 (Airmail)**
☐ **2 Years: UK £94.99, Europe £109.99 (Airmail), Rest of the World £135.99 (Airmail)**

To: Practical Electronics
Wimborne Publishing Ltd
113 Lynwood Drive, Wimborne, Dorset BH21 1UU
Tel: 01202 880299 Fax: 01202 843233
Email: stewart.kearn@wimborne.co.uk

I enclose payment of £ ....
(cheque/PO in £ sterling only)
payable to *Practical Electronics*

☐ Please charge my Visa/Mastercard

My card number is: ....
Please print clearly, and check that you have the number correct

Card Security Code .... Valid From Date........................
(The last 3 digits on or just under the signature strip)

Card Ex. Date ....

Name ....

Address ....

Post code .... Tel. ....

Email ....

Subscriptions start with the next available issue.

World-Class Analog from a Microcontroller Leader?

If You Only Know Microchip as an MCU Supplier, We're About to Blow Your Mind

Microchip's success story wouldn't be complete without including our analog solutions. Our history as a leading solution supplier providing comprehensive design support and a broad product portfolio doesn't only include our microcontroller products.

We also offer high-performance, easy-to-implement linear, mixed-signal, power management, thermal and interface products. When combined, Microchip's extensive portfolio can be used in numerous applications with various performance requirements.

You will have the power, flexibility and confidence to choose the right solution for your design, regardless of design constraints. Take advantage of our experience and complete system solutions to save time and simplify your design effort.

Start your own success story at www.microchip.com/Real-Analog

MICROCHIP

The Microchip name and logo and the Microchip logo are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries. All other trademarks are the property of their registered owners.
© 2019 Microchip Technology Inc. All rights reserved. DS20006063A. MEC2243A-ENG-08-19

WIRELESS FOR THE WARRIOR

by LOUIS MEULSTEE

THE DEFINITIVE TECHNICAL HISTORY OF RADIO COMMUNICATION EQUIPMENT IN THE BRITISH ARMY

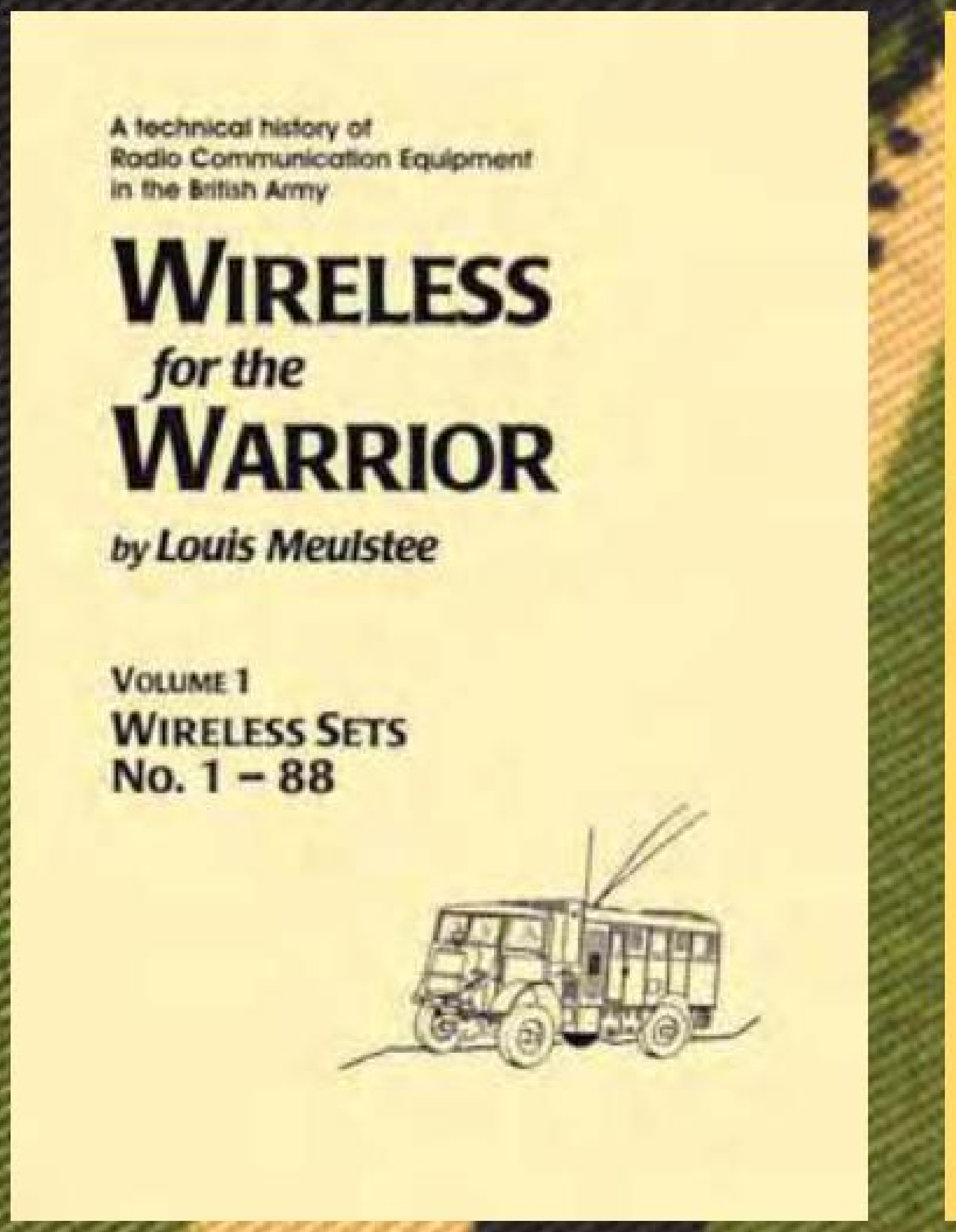
The *Wireless for the Warrior* books are a source of reference for the history and development of radio communication equipment used by the British Army from the very early days of wireless up to the 1960s.

The books are very detailed and include circuit diagrams, technical specifications and alignment data, technical development history, complete station lists and vehicle fitting instructions.

Volume 1 and *Volume 2* cover transmitters and transceivers used between 1932-1948. An era that starts with positive steps taken to formulate and develop a new series of wireless sets that offered great improvements over obsolete World War I pattern equipment. The other end of this timeframe saw the introduction of VHF FM and hermetically sealed equipment.

Volume 3 covers army receivers from 1932 to the late 1960s. The book not only describes receivers specifically designed for the British Army, but also the Royal Navy and RAF. Also covered: special receivers, direction finding receivers, Canadian and Australian Army receivers, commercial receivers adopted by the Army, and Army Welfare broadcast receivers.

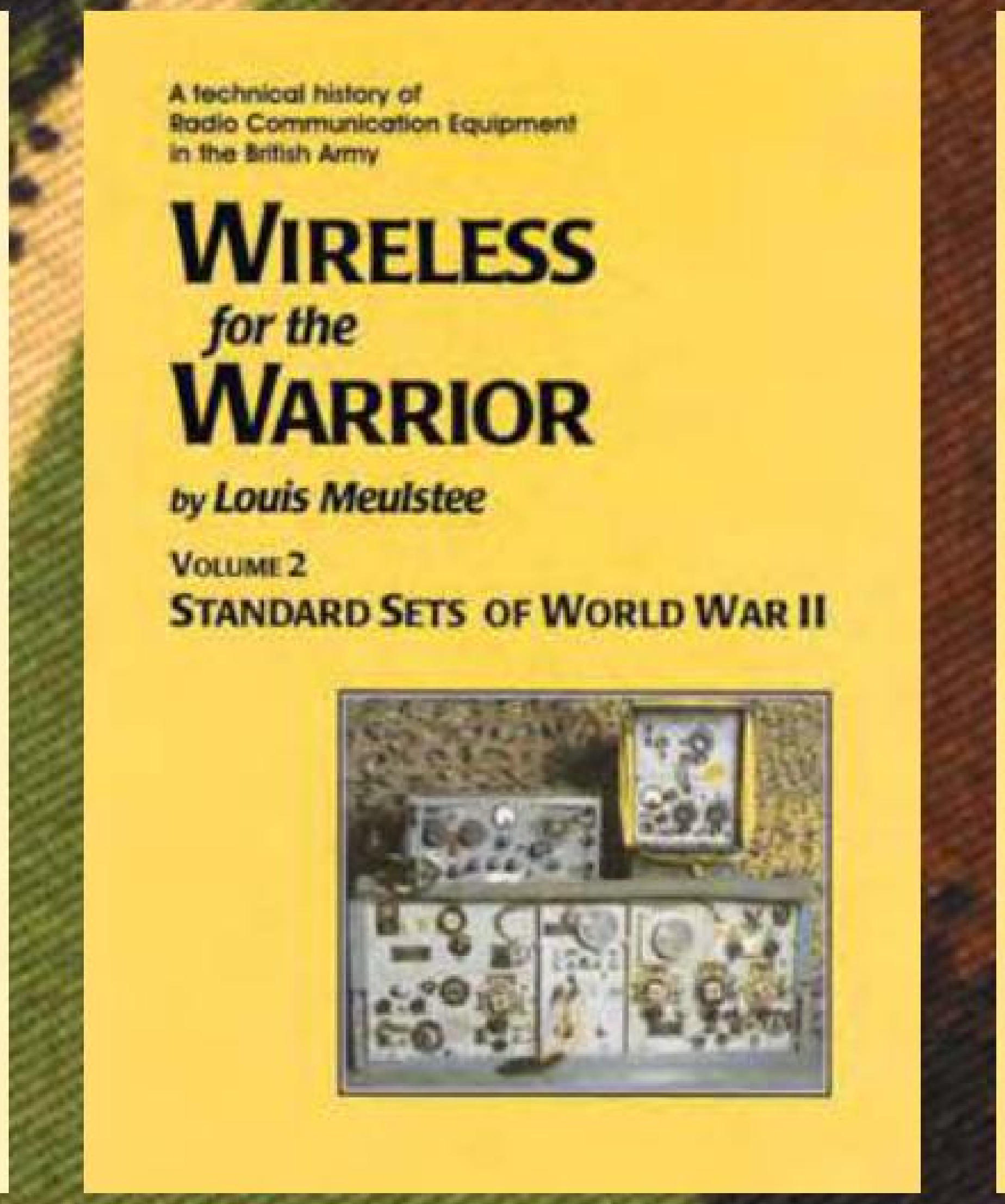
Volume 4 covers clandestine, agent or 'spy' radio equipment, sets which were used by special forces, partisans, resistance, 'stay behind' organisations, Australian Coast Watchers and the diplomatic service. Plus, selected associated power sources, RDF and intercept receivers, bugs and radar beacons.



A technical history of
Radio Communication Equipment
in the British Army

WIRELESS
for the
WARRIOR
by Louis Meulstee

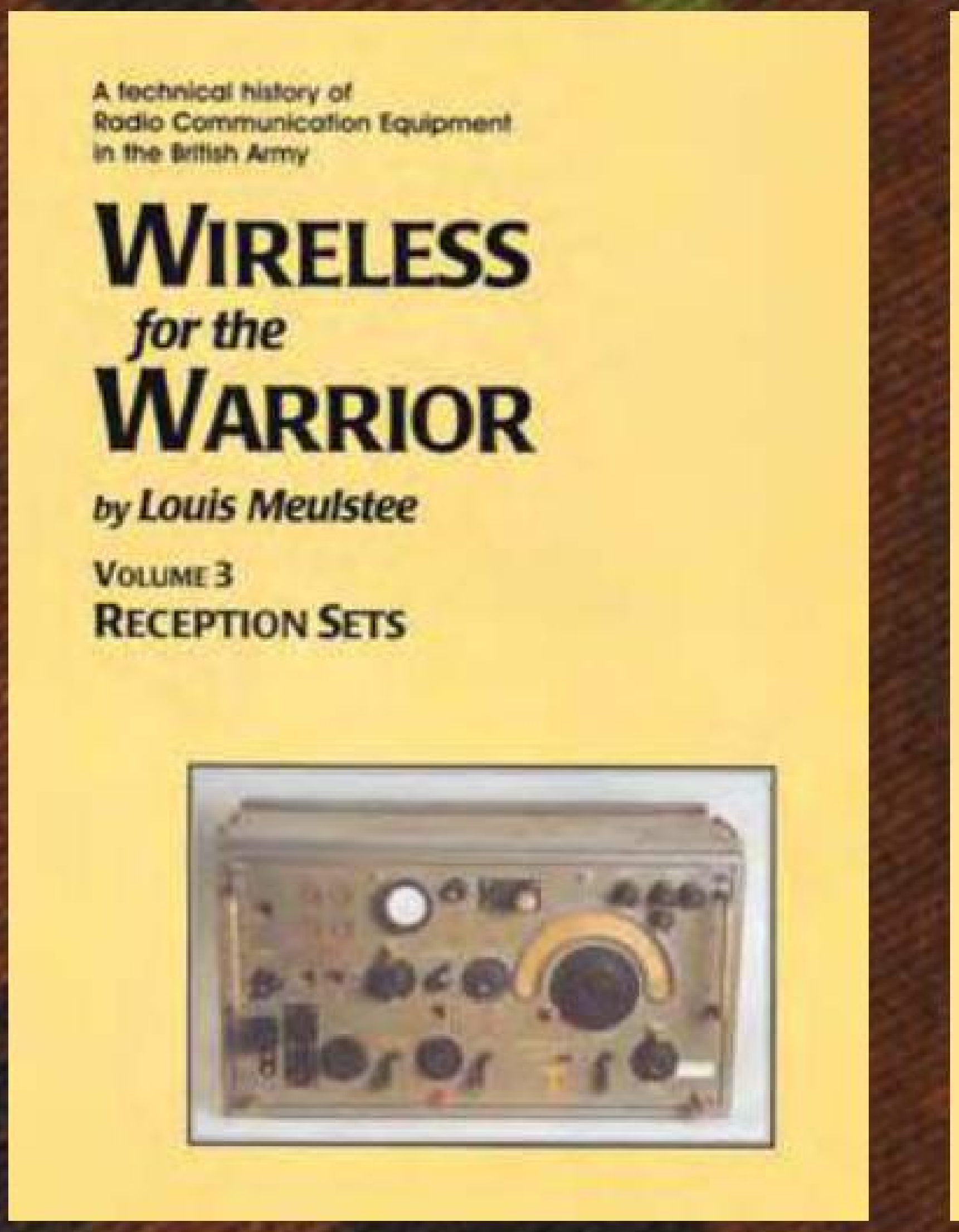
Volume 1
Wireless Sets
No. 1 – 88



A technical history of
Radio Communication Equipment
in the British Army

WIRELESS
for the
WARRIOR
by Louis Meulstee

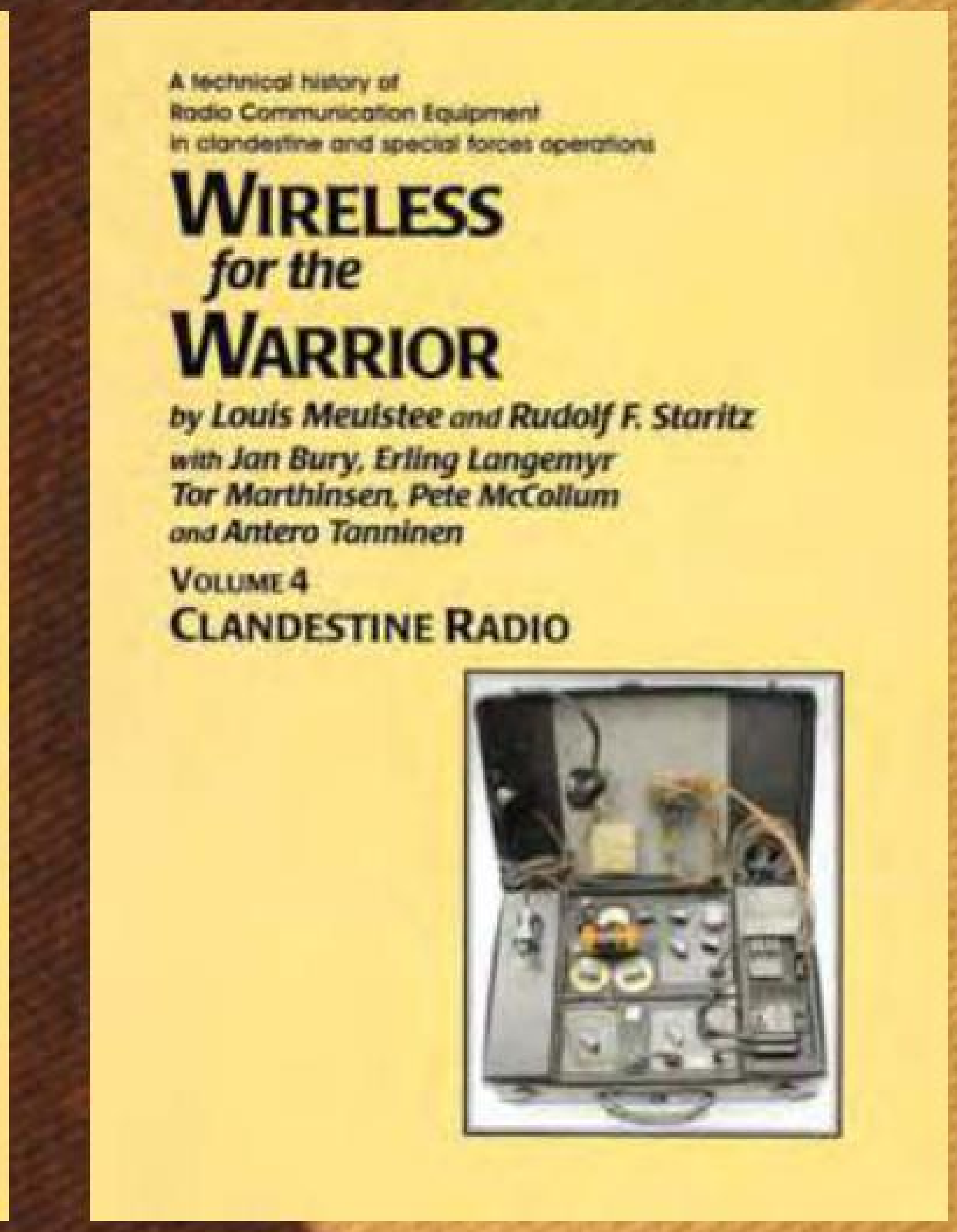
Volume 2
Standard Sets of World War II



A technical history of
Radio Communication Equipment
in the British Army

WIRELESS
for the
WARRIOR
by Louis Meulstee

Volume 3
Reception Sets



A technical history of
Radio Communication Equipment
in clandestine and special forces operations

WIRELESS
for the
WARRIOR
by Louis Meulstee and Rudolf F. Staritz
with Jan Bury, Erling Langemyr
Tor Marthinsen, Pete McCollum
and Antero Tanninen

Volume 4
Clandestine Radio

ORDER YOURS TODAY!
JUST CALL 01202 880299 OR VISIT www.epemag.com

Practical Electronics

Volume 48. No. 11
November 2019
ISSN 2632 573X

Editorial

Editorial offices

Practical Electronics
Electron Publishing Limited
1 Buckingham Road
Brighton
East Sussex BN1 3RA

Tel 01273 777619
Mob 07973 518682
Fax 01202 843233
Email pe@electronpublishing.com
Web www.epemag.com

Advertisement offices

Practical Electronics Adverts
1 Buckingham Road
Brighton
East Sussex BN1 3RA

Phone 01273 777619
Mob 07973 518682
Email pe@electronpublishing.com

Editor Matt Pulzer
General Manager Louisa Pulzer
Subscriptions Stewart Kearn
Online Editor Alan Winstanley
Web Systems Kris Thain
Publisher Matt Pulzer

Technical enquiries

We regret technical enquiries cannot be answered over the telephone. We are unable to offer any advice on the use, purchase, repair or modification of commercial equipment or the incorporation or modification of designs published in the magazine. We cannot provide data or answer queries on articles or projects that are more than five years old.

Questions about articles or projects should be sent to the editor by email: pe@electronpublishing.com

Projects and circuits

All reasonable precautions are taken to ensure that the advice and data given to readers is reliable. We cannot, however, guarantee it and we cannot accept legal responsibility for it.

A number of projects and circuits published in *Practical Electronics* employ voltages that can be lethal. You should not build, test, modify or renovate any item of mains-powered equipment unless you fully understand the safety aspects involved and you use an RCD (GFCI) adaptor.

Component supplies

We do not supply electronic components or kits for building the projects featured, these can be supplied by advertisers. We advise readers to check that all parts are still available before commencing any project in a back-dated issue.

Advertisements

Although the proprietors and staff of *Practical Electronics* take reasonable precautions to protect the interests of readers by ensuring as far as practicable that advertisements are bona fide, the magazine and its publishers cannot give any undertakings in respect of statements or claims made by advertisers, whether these advertisements are printed as part of the magazine, or in inserts. The Publishers regret that under no circumstances will the magazine accept liability for non-receipt of goods ordered, or for late delivery, or for faults in manufacture.

Transmitters/bugs/telephone equipment

We advise readers that certain items of radio transmitting and telephone equipment which may be advertised in our pages cannot be legally used in the UK. Readers should check the law before buying any transmitting or telephone equipment, as a fine, confiscation of equipment and/or imprisonment can result from illegal use or ownership. The laws vary from country to country; readers should check local laws.

Downloading files

At least once a week, we receive an email along the lines of this one from reader Les Hewett: ‘I recently purchased a 15-year DVD bundle from you, which I am very pleased with. However, many of the construction articles require a PIC microcontroller programmed with a hex file. I have not been able to find these files on your website; however, I have found them listed on *Silicon Chip* magazine, where the original article appeared. Is there any arrangement by which the required files can be downloaded without taking out a subscription to *Silicon Chip*? Since many of the construction projects make use of microcontrollers, the usefulness of your DVDs is limited if I am not able to download the required software.’

The short answer is definitely, ‘yes’ – we do provide the files, and without further charge. So, where are they?

The email from Les asks about *Silicon Chip* software, but reader questions also apply to *Circuit Surgery* simulation files, *Audio Out* PCB files or any other useful download mentioned in *PE* articles and projects. Eventually, it dawned on us that readers of the finest magazine devoted to electronics – a topic that requires intelligence, persistence, troubleshooting skills and more than a little experience of using the Internet – were not the problem. We’re the problem, or to be specific, the less than obvious location of the files. So, first of all, apologies to anyone who has ever wasted time looking for files. Here is an explanation of what is where and how you find it.

In theory, most of the code and all other file types, including *Silicon Chip* project code is available without charge on the *PE* website. You can find it by clicking on the main ‘Library’ tab and going to: **https://www.epemag3.com/library.html**

There, just choose a year/issue and click the link; then, under the image of that issue’s cover, click on the ‘Source code’ zip file link – for example, **0214.zip** for February 2014. This will give you all the relevant files (not just code) for a particular issue. Most of the time PIC code will be there; however, do note four possible exceptions:

i) We are not allowed to distribute the code and you have to buy a pre-programmed PIC from whoever supplied the project. This happens rarely, and we’ll point you to the PIC vendor in the article.

ii) In a multi-part project you may find that the software is contained in the zip file for an earlier or later month.

iii) Occasionally, we fail to upload the software – just chase us!

iv) When you program a Microchip PIC with a PIC kit using Microchip software there are various optimisation levels. The base-level free software is good, but the files it generates take more space than the non-free versions. If the free version generates more machine code than can fit in the PIC’s Flash ROM then the non-free software is used and vendors generally assume that readers will not have access to the paid-for compilers so the code is not distributed. Again, in this rare scenario you need to buy a pre-programmed PIC.

To be clear, almost all code and other files are available, but we cannot guarantee the availability of everything.

Matt Pulzer
Publisher

NEWS

A roundup of the latest news from the world of electronics and technology

Meet the OU's earliest computers

The Open University (OU) is the largest academic institution in the UK and a world leader in flexible distance learning. Celebrating its 50th anniversary in 2019, the university was founded with a clear purpose: to open up education to all. Since those beginnings in 1969, the OU has taught more than two million students worldwide and has over 170,000 current students, including more than 15,000 overseas.

To mark the 50th anniversary of the OU, a new display celebrating its early technology is open to visitors to The National Museum of Computing at Bletchley Park until 31 December 2019.

As pioneers in distance learning education, before personal computers were the norm and the Internet became commonplace, the OU developed several innovative methods to enable its students to study on computers in their own homes. The exhibition offers a rare opportunity to see six of these machines. It also presents an OU timeline, a video about the OU's early computing and micro-electronics courses and an array of historic photographs of the OU's computing since its foundation in 1969.

Roger Moore, who worked on the OU Home Computing Policy Unit in the 1980s and in various OU educational software roles until 2015, highlights the significance of the OU's computing education: 'In 1972, vice-chancellor Walter Perry, noted that the OU had turned out 2,000 students capable of writing non-trivial BASIC programmes. Now, 50 years on, there are probably more than one million OU graduates who have gone on to advance the development of computing and IT in this country.'

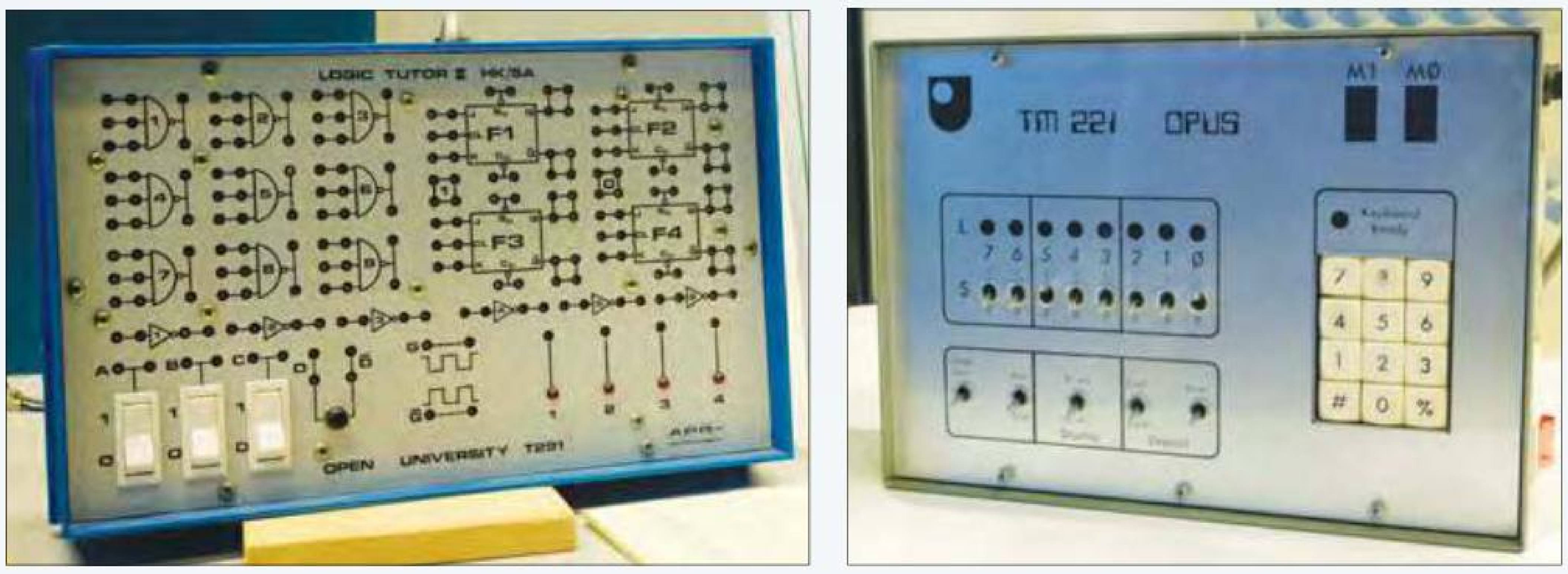
The display has been curated by Roger Moore, a TNMOC volunteer, with OU colleagues past and present. They have researched and brought together six of the early OU machines, including the Logic Tutor, MICRO 1, HEKTOR, OPUS and DESMOND, with some of the course software. Restoration of a HEKTOR is underway with the aim of giving visitors an opportunity for hands-on access to the early days of OU computing, and to program in Assembler or BASIC.

With three national computer centres – in Milton Keynes, London and Newcastle-upon-Tyne – the OU grew to have up to 275 study centres across the country with computer access to the centres. The HEKTORs also enabled some students access to the computer centres from home.

Regarded as the UK's major e-learning institution, the OU is a world leader in developing technology to increase access to education on a global scale. For further information please visit: **www.open.ac.uk**

The National Museum of Computing

The National Museum of Computing, located on Bletchley Park in Block H, one of England's 'irreplaceable places', is an independent charity housing the world's largest collection of functional historic computers, including reconstructions of the wartime code-breaking Colossus and the Bombe, and the WITCH, the world's oldest working digital computer. The Museum enables visitors to follow the development of computing from the ultra-secret pioneering efforts of the 1940s through the large systems and mainframes of the 1950s, 60s and 70s, and the rise of personal computing in the 1980s and beyond. For more information, see: **www.tnmoc.org**



Early OU machines for teaching computing: (left) Logic Tutor from 1974, designed to show the basics of gates and flip-flops; and (right) An OPUS microcomputer, which had an Intel 8049 processor (clock speed 1MHz), 128 bytes of RAM and 2048 bytes of ROM.

HAMMOND MANUFACTURING®

Hand held enclosures standard & waterproof

www.hammondmfg.com/1553.htm
www.hammondmfg.com/1553W.htm

01256 812812
sales@hammond-electronics.co.uk

PCBWay

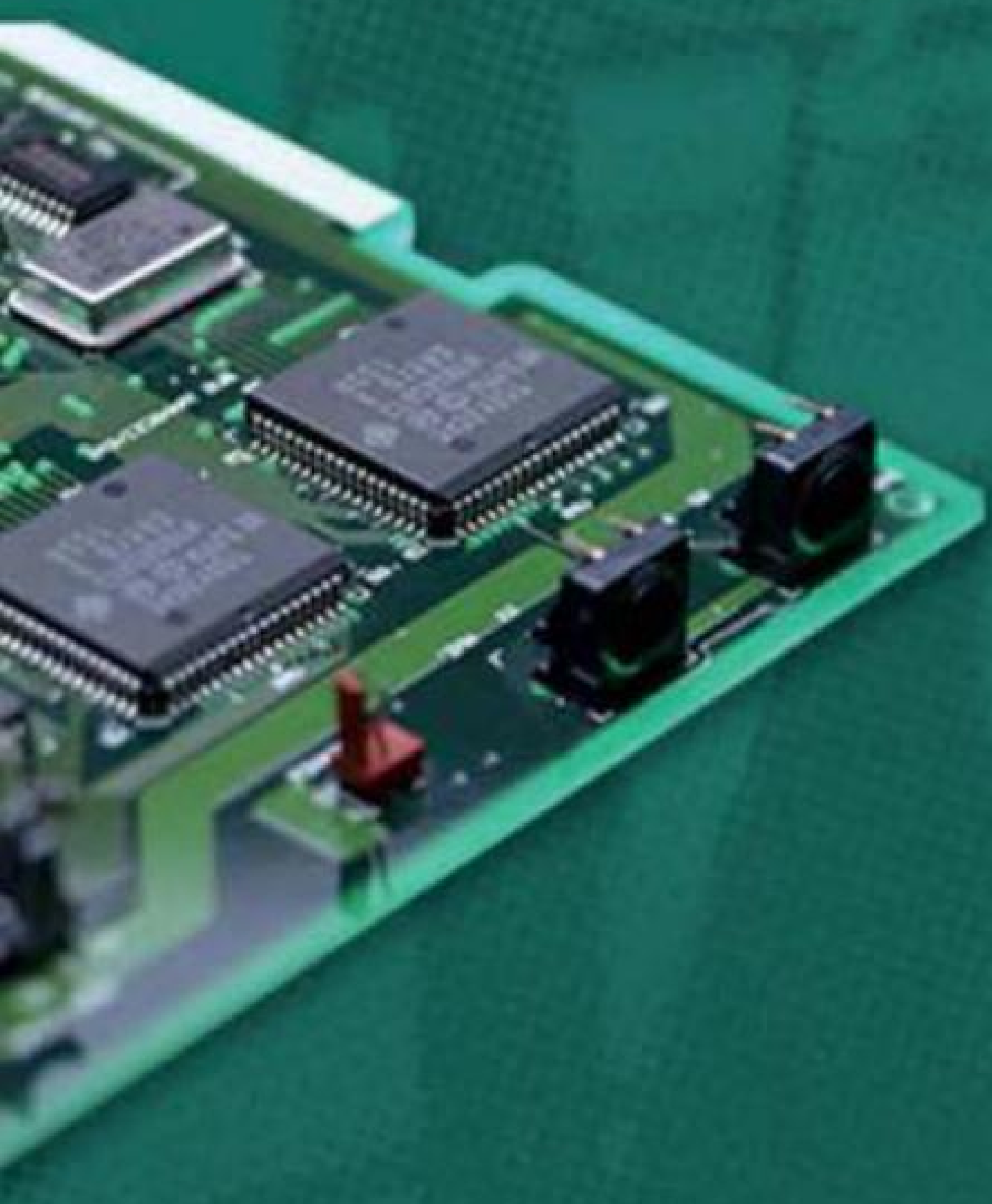
Full-feature custom PCB manufacturing service

CUSTOM PCB SERVICE

Only $5

for 10 boards

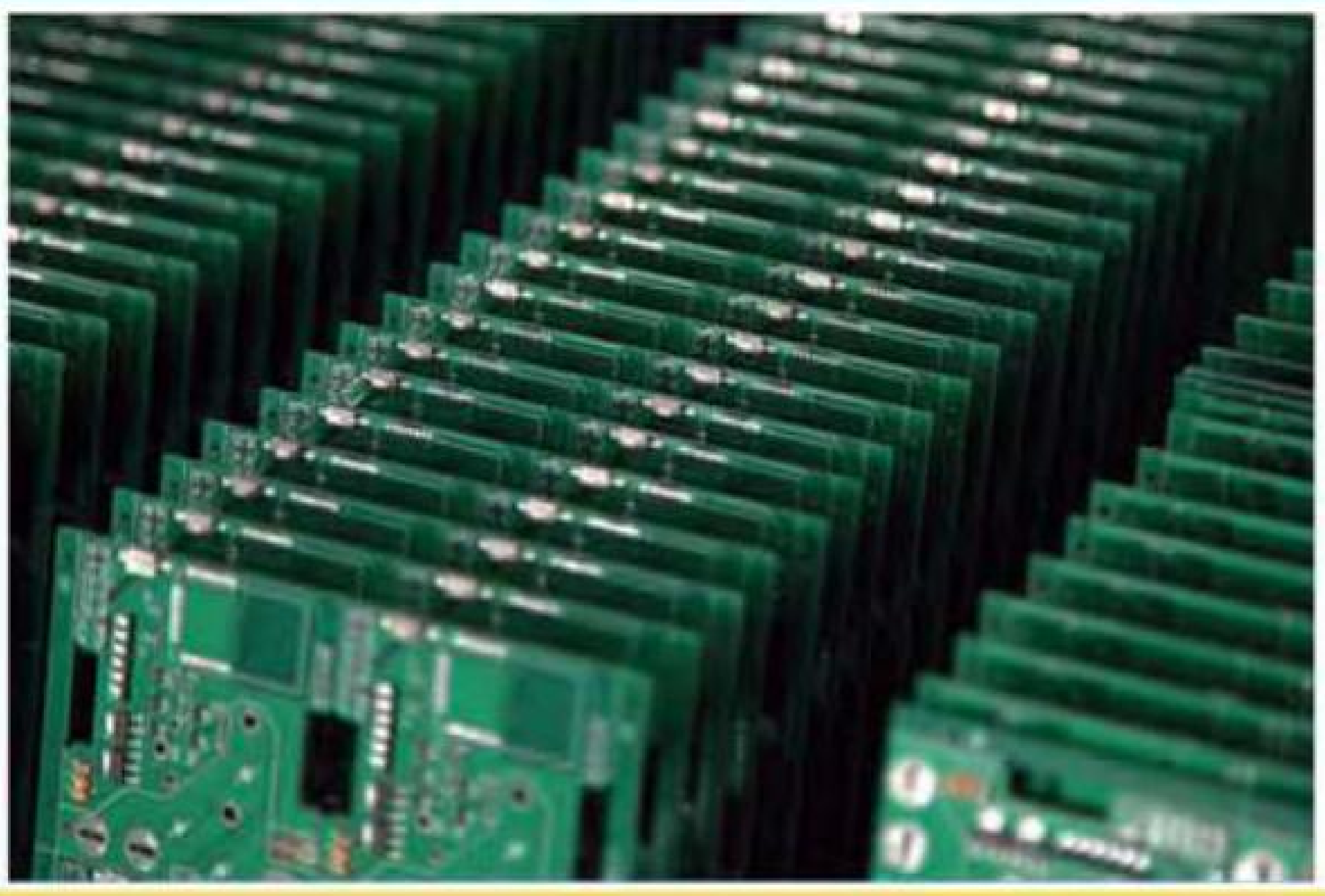
- ★ Rogers, HDI, aluminum and rigid-flex PCB are available now
- ★ 24-hour production time
- ★ Best punctuality record



SMT & THT ASSEMBLY

Start from $30

- ★ Free shipping + Free stencil
- ★ Component sourcing
- ★ Quality assurance



WEBSITE
WWW.PCBWay.com

Get an instant quote online: www.pcbway.com

24/7 customer service: service@pcbway.com

Verified by:
ISO IAF UL

Greater flexibility

Techno Talk

Mark Nelson

That's what we need. Not just in batteries, but also in our national electric power generation capacity. These are the topics for this month's sermon, which may well surprise you in several respects.

It's a real nightmare. You spend three weeks designing a hobby electronics project, laying out and making a PCB, fitting the components and drilling the enclosure. Finally, you put it all together and only then do you find that the lid of the case won't fit, because the power supply is 5mm too tall!

For a hobbyist this is frustrating, but not quite a catastrophe. For commercial manufacturers it's far worse. Back in the 1950s, a well-known British manufacturer of broadcast equipment developed their new TV camera for studio and outside broadcast use and discovered, only at the last minute, that the mains transformer did not fit the space allotted on the jig-drilled chassis. The mass of other components that occupied the rest of the chassis could not be squeezed any closer together, so what could be done? Amazingly the chosen solution was to shrink the mains transformer in a hydraulic press – 'desperate situation calls for desperate solution'.

Initially, this ruse worked, but soon reports came in of cameras catching fire during TV programmes. Disastrous!

Not only bendy, but also stretchy

Back to the present, where the electronicists of tomorrow could have an alternative solution. Not quite a shape-shifting transformer, but a 're-dimensionable' battery. Yes, we've all heard about bendy batteries for wearables and biological implants, but with the advent of *stretchable* batteries those are now old hat. *Physics World* magazine reports that conductive layers that stay conducting even when stretched by as much as 300% in any direction could find use in a new generation of deformable and wearable electronic devices.

A team of researchers led by Nicholas Kotov of the University of Michigan in the US, working with colleagues in South Korea, has now developed stretchable conductors using a new type of film made from multiple layers of plastic polymers and gold nanoparticles. To test their new material, they used it as the electrode in a lithium-ion battery. The result could allow for the development of stretchable lithium-ion batteries that deliver stable power even when deformed, they say.

'These materials – and stretchable devices made from them – open the road to a new generation of implantable biomedical devices with integrated batteries that are able to deform in a variety of directions inside the body. They could thus be employed in constrained environments inside organs, for example, and perhaps even in the vicinity of neurons. They might also be used in soft robotics applications', explained Kotov.

Power stability

By the time you read this, last August's 'Great British Blackout', in which nearly a million homes, hospitals and other users had their power turned off, should be but a dim memory, but it still raises awkward questions about the stability of our National Grid infrastructure. The reason given for disconnecting power to the unfortunate customers was the inability to assure the correct mains frequency of 50Hz, rather than maintain the correct voltage of 230V +10% or –6%. The National Grid is obliged by its licence to control the frequency within ±1% of 50Hz so it can fluctuate between 49.5Hz to 50.5Hz. (In fact, normal operational limits are 50Hz ±0.4%.)

But why is frequency more critical than voltage? The Drax power station website explains that maintaining a consistent frequency is critical because multiple frequencies do not mix without damaging equipment. Thus, every generator (gas, coal, nuclear, solar or wind-powered) in England, Scotland and Wales connected to the high-voltage transmission system is synchronised to every other generator.

How is frequency managed?

If the volume generated does not match the exact amount of electricity being used, it can affect the frequency of the electricity on the grid. If there's more demand for electricity than there is supply, frequency will fall. If there is too much supply, frequency will rise, and there's a very slim margin of error. The Grid instructs power generators to make their generating units respond automatically to changes in frequency. For traditional generators this is simple and can be handled in seconds. If the frequency rises, the turbine reduces its steam inflow. If it falls, it will increase, changing the electrical output.

With solar and wind-powered generators the challenge is greater. It's possible to regulate wind output down or hold back wind turbines to enable upward frequency response when there is sufficient wind. Similarly, solar panels can be switched on and off to simulate frequency response, although it is less simple for the National Grid to instruct and monitor them. Existing nuclear power stations cannot regulate frequency at all.

Potential solution

What we need then is more generating power, assuming we cannot reduce demand. The comparative costs for power generation are changing, with the balance favouring onshore wind and solar power in all major economies except for Japan. Consequently, the continuous development of onshore wind farms is an important feature of the European transition towards an energy system powered by distributed renewables.

The good news, according to researchers from the universities of Sussex and Aarhus, is that nearly half of Europe's undeveloped landmass is theoretically suitable for onshore wind turbines. Even better, if all of these potential sites were utilised, they could generate 497EJ exajoules of power, exceeding the total *global* energy demand forecast for 2050 (one exajoule (EJ) equals 10^{18} joules).

'Critics will no doubt argue that the naturally intermittent supply of wind makes onshore wind energy unsuitable to meet the global demand,' said Peter Enevoldsen, assistant professor at Aarhus University's Centre for Energy Technologies. 'But even without accounting for new developments in wind turbine technology, onshore wind power is the cheapest mature source of renewable energy, and utilising the different wind regions in Europe is the key to meet the demand for a 100% renewable and fully decarbonised energy system."

STORE YOUR BACK ISSUES ON CD-ROMS

VOL 1	**BACK ISSUES**	– January 1999 to June 1999 Plus bonus material from Nov and Dec 1998
VOL 2	**BACK ISSUES**	– July 1999 to December 1999
VOL 3	**BACK ISSUES**	– January 2000 to June 2000
VOL 4	**BACK ISSUES**	– July 2000 to December 2000
VOL 5	**BACK ISSUES**	– January 2001 to June 2001
VOL 6	**BACK ISSUES**	– July 2001 to December 2001
VOL 7	**BACK ISSUES**	– January 2002 to June 2002
VOL 8	**BACK ISSUES**	– July 2002 to December 2002
VOL 9	**BACK ISSUES**	– January 2003 to June 2003
VOL 10	**BACK ISSUES**	– July 2003 to December 2003
VOL 11	**BACK ISSUES**	– January 2004 to June 2004
VOL 12	**BACK ISSUES**	– July 2004 to December 2004
VOL 13	**BACK ISSUES**	– January 2005 to June 2005
VOL 14	**BACK ISSUES**	– July 2005 to December 2005
VOL 15	**BACK ISSUES**	– January 2006 to June 2006
VOL 16	**BACK ISSUES**	– July 2006 to December 2006
VOL 17	**BACK ISSUES**	– January 2007 to June 2007
VOL 18	**BACK ISSUES**	– July 2007 to December 2007
VOL 19	**BACK ISSUES**	– January 2008 to June 2008
VOL 20	**BACK ISSUES**	– July 2008 to December 2008
VOL 21	**BACK ISSUES**	– January 2009 to June 2009
VOL 22	**BACK ISSUES**	– July 2009 to December 2009

ONLY £16.45 each
Five years just £29.95
including VAT and p&p

VOL 23	**BACK ISSUES**	– January 2010 to June 2010
VOL 24	**BACK ISSUES**	– July 2010 to December 2010
VOL 25	**BACK ISSUES**	– January 2011 to June 2011
VOL 26	**BACK ISSUES**	– July 2011 to December 2011
VOL 27	**BACK ISSUES**	– January 2012 to June 2012
VOL 28	**BACK ISSUES**	– July 2012 to December 2012
VOL 29	**BACK ISSUES**	– January 2013 to June 2013
VOL 30	**BACK ISSUES**	– July 2013 to December 2013
VOL 31	**BACK ISSUES**	– January 2014 to June 2014
VOL 32	**BACK ISSUES**	– July 2014 to December 2014
VOL 33	**BACK ISSUES**	– January 2015 to June 2015
VOL 34	**BACK ISSUES**	– July 2015 to December 2015
VOL 35	**BACK ISSUES**	– January 2016 to June 2016
VOL 36	**BACK ISSUES**	– July 2016 to December 2016
VOL 37	**BACK ISSUES**	– January 2017 to June 2017
VOL 38	**BACK ISSUES**	– July 2017 to December 2017
VOL 39	**BACK ISSUES**	– January 2018 to June 2018
VOL 40	**BACK ISSUES**	– July 2018 to December 2018

FIVE-YEAR DVD/CD-ROMs

No.1 – Jan 03 to Dec 07
No.2 – Jan 04 to Dec 08
No.3 – Jan 05 to Dec 09
No.4 – Jan 06 to Dec 10
No.5 – Jan 07 to Dec 11
No.6 – Jan 08 to Dec 12
No.7 – Jan 09 to Dec 13
No.8 – Jan 10 to Dec 14
No.9 – Jan 11 to Dec 15
No.10 – Jan 12 to Dec 16
No.11 – Jan 13 to Dec 17
No.12 – Jan 14 to Dec 18 – NEW!

15 YEARS OF EPE!

AMAZING BUNDLE OFFER

No.1 – Jan 03 to Dec 07
PLUS
No.6 – Jan 08 to Dec 12
PLUS
No.11 – Jan 13 to Dec 17

ALL FOR JUST
£39.95

All DVD/CD-ROMs are suitable for any PC with a DVD/CD-ROM drive.
They require Adobe Acrobat Reader (free from: www.adobe.com/acrobat)

ORDER YOUR CD-ROMs TODAY!
JUST CALL 01202 880299 – OR VISIT www.epemag.com

Net Work

Alan Winstanley

Smile! You're on candid camera… well, in fact it's no joking matter. This month, we look at the extraordinary ambition of surveillance technology – from number plates to lip reading.

Britain is being carpeted by CCTV coverage, comprising both privately owned cameras and those operated by local authorities. It's estimated that Britain now hosts one surveillance camera for every 32 of its citizens.

Ordinary folk going about their everyday life can expect to be captured on camera dozens of times. In central London it's reckoned that over 600,000 cameras monitor footfall, or one camera for every 14 Londoners. Someone taking a one-hour lunch break may be snapped on the street as many as 300 times, the commercial website **CCTV.co.uk** says.

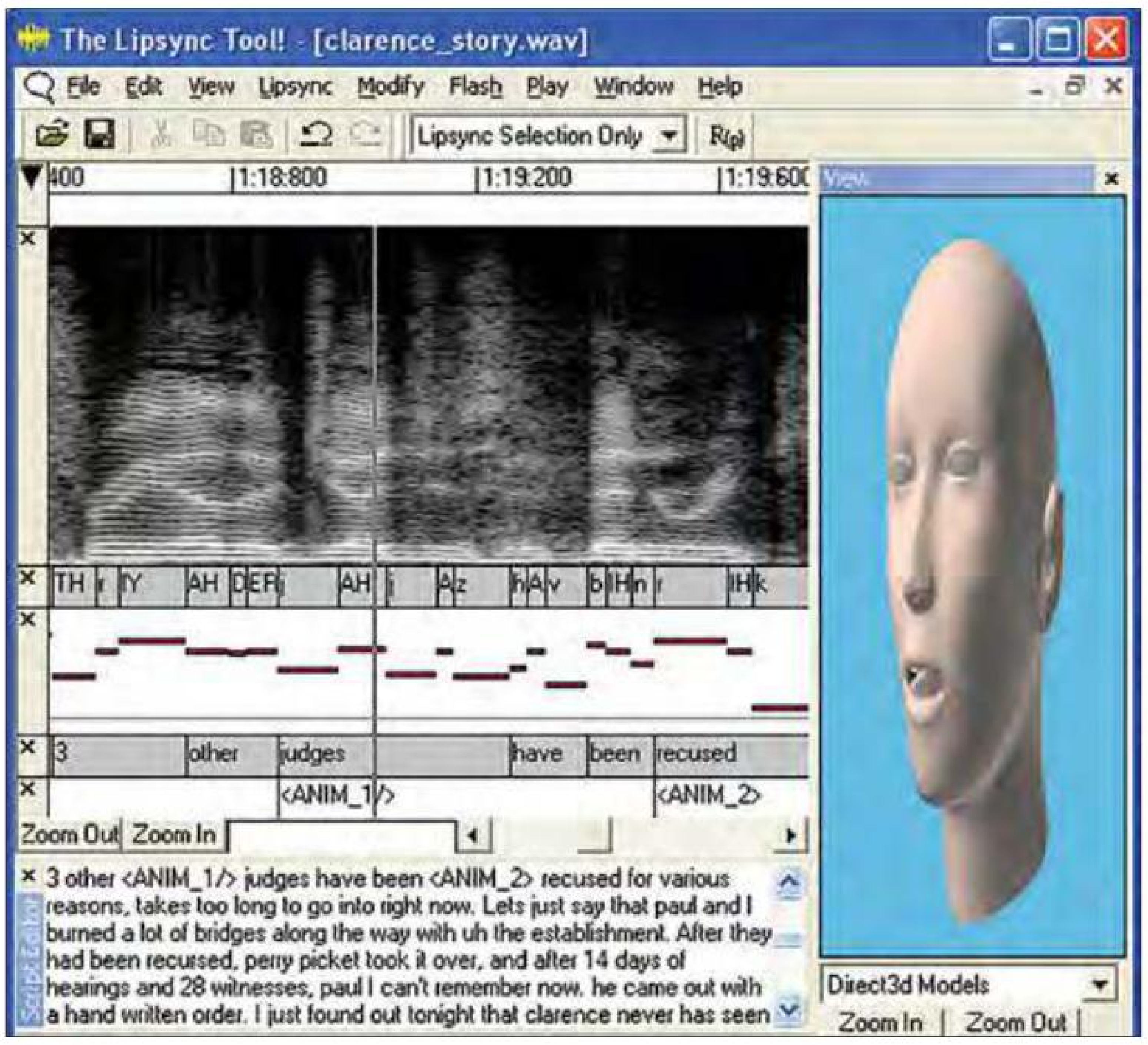
Many businesses are 'cabling up' disk-based CCTV packages on their premises, and cloud-based products like the Google Nest Cam offer domestic users remote monitoring of property with images beamed to their smartphone or tablet. Such is the widespread use of CCTV that miscreants have taken to obscuring themselves from head to foot in order to make video evidence unusable.

The problem of 'reading' something like a document, image or CCTV video recording, and intelligently extracting information from it is a thorny one, and electronic systems that interpret and process such data 'artificially' have been around a long time. Automating these processes brings more reliability, speed and efficiency rather than relying on error-prone and tired humans to handle such onerous tasks.

Making data useable

Nearly half a century ago, computers were already using magnetic ink character recognition (MICR) to scan the special account characters imprinted along the bottom of cheques, which automated the processing of cheque payments. Optical mark recognition (OMR) on lottery tickets or multiple-choice questionnaires, optical character recognition (OCR) of letters and correspondence, or keyword scanning of job applications are techniques commonly used to capture and analyse information. During my days at the helm of our *Ingenuity Unlimited* column, and before the advent of email, I asked for typed correspondence, which allowed me to OCR the contributor's text and scan it to disk. The accuracy of Scansoft's software (now called Kofax OmniPage) was already very high and it eliminated the drudgery of retyping the copy and making errors (mine).

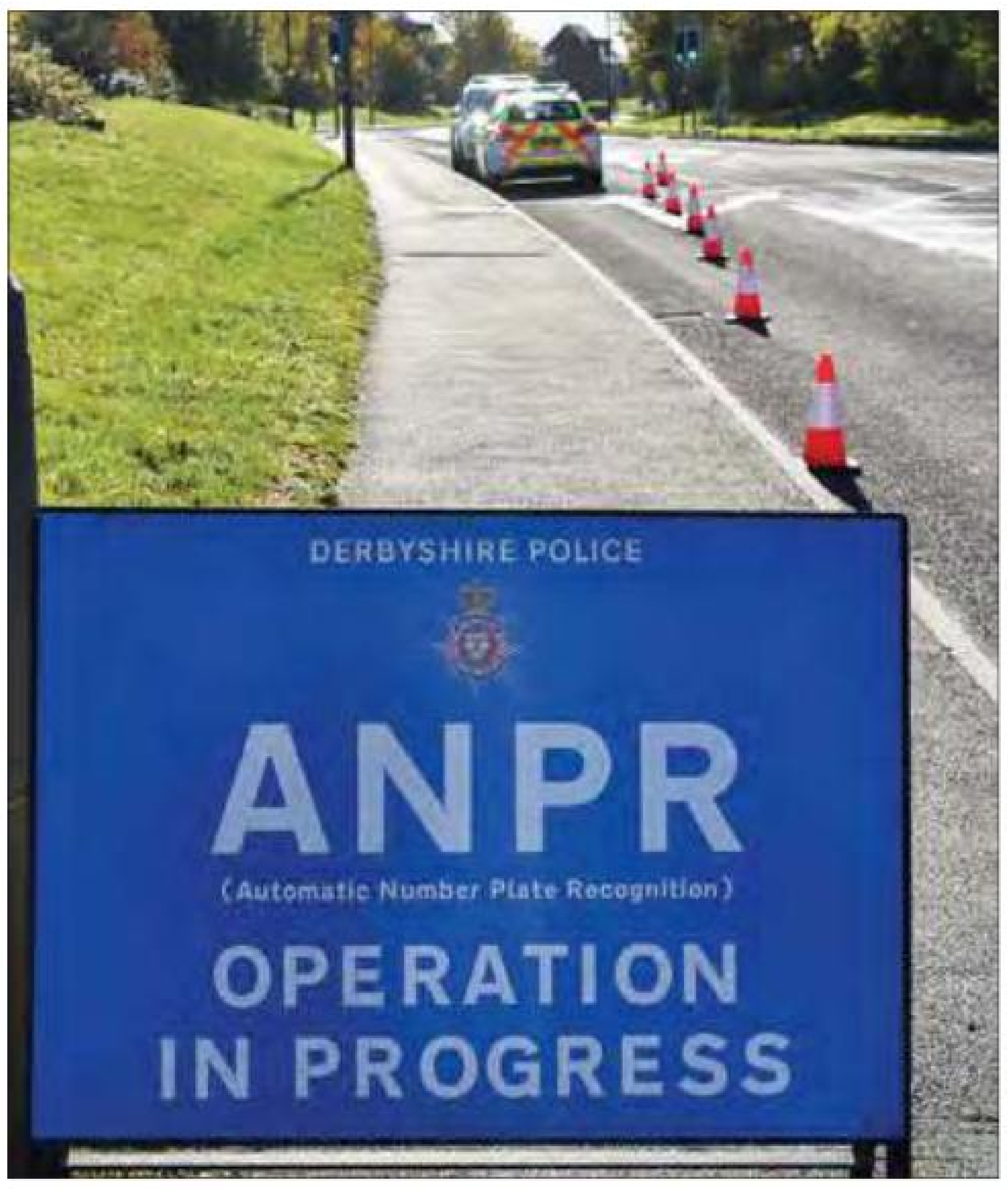
So the idea of capturing text or images and trying to interpret them 'artificially' using machine intelligence is nothing new. More than 20 years ago, our '*Surgery*' writer Ian Bell explained to the writer an ambitious project by one of his students, which endeavoured to recognise beetles or bugs that passed underneath a digital camera. The idea was that the critter would be lit from underneath using an electroluminescent panel, and the resulting bitmap image could be referenced against a look-up table – very impressive stuff at the time.



3 other <ANIM_1/> judges have been <ANIM_2> recused for various reasons, takes too long to go into right now. Lets just say that paul and I burned a lot of bridges along the way with uh the establishment. After they had been recused, perry picket took it over, and after 14 days of hearings and 28 witnesses, paul I can't remember now. he came out with a hand written order. I just found out tonight that clarence never has seen

The free version of Lipsync shows how speech tracks can be converted into phonemes prior to mapping them to 'visemes'.

Technology has raced ahead, and a common application today of automated video capture and data processing is roadside or mobile ANPR (automatic number plate recognition) cameras that scan fixed-format licence plates and alert control rooms or police cars when an offending vehicle 'pings' the system. As we would expect, ANPR technology has crept out sideways in our society and ANPR systems are cropping up even in benign environments like council-operated 'recycling centres' (garbage dumps): ANPR can scan number plates to flag up 'outsider' vehicles not registered within a local borough and close the gates automatically to prevent vehicles from proceeding until the driver has been checked out. The use of ANPR to control supermarket and hospital car parks is something that plagues British motorists constantly, especially when they receive a fine in the mail two weeks later for overstaying their allotted time. Mainland European police forces do the same, fining UK motorists many months if not years after 'snapping' a car speeding while abroad.



An invasion of privacy? Smart systems like ANPR are spreading across public spaces.

If the face fits

On the consumer front, webcam and smartphone AI can already superimpose cartoon characters on facial features for a bit of fun, and ‘smile recognition’ lets cameras take the best photos of a smiling subject. Facebook can optionally ‘tag’ facial images to ‘create a better experience’ as Facebook puts it (a feature that Facebook is looking to disable by default, at long last).

Back to CCTV, and observers claim that our privacy faces ‘extinction’ as the country is in the grip of a ‘facial recognition epidemic’. Machine intelligence is being used to ‘read’ our facial biometrics automatically, and covert trials of this form of monitoring have already taken place in major UK shopping malls and other public amenity areas involving the police and millions of unsuspecting subjects. A small number of prisons have also tried scanning visitors’ biometrics, sometimes reducing the number of visits from certain persons who knew they were being ‘scanned’ this way.

Whether ANPR or automated facial recognition (AFR), it has been interesting to see our society sleep-walking into accepting these levels of automation and artificial intelligence in everyday life. Imagine hooking AFR cameras in real time to a national database of, say, passport photographs or crime records. The implementation of AFR is new territory, and Britain’s Information Commissioner’s Office is belatedly investigating the impact on an individual’s privacy that the technology may have. Inevitably, as this technology edges outwards it will routinely appear at airports, railway stations, border crossings and many other public places. Privacy group Big Brother Watch in its ‘Face Off’ campaign claims that, in limited trials, 95% of police facial recognition ‘matches’ wrongly identified innocent people, so clearly this technology still has some way to go. The Big Brother Watch report *Face Off: The lawless growth of facial recognition in UK policing* is available at: **http://bit.ly/pe-nov19-bb**

Read my lips

There’s more to come: mainstream media and pundits have taken to reporting topical ‘private’ comments or conversations caught on film, but out of earshot, by employing lip readers to analyse what’s being said, to the embarrassment of a number of public figures, politicians and footballers. Lip sync software is being developed that does the same job powered by artificial intelligence: in 2017 the IEEE published a paper, *Lip Reading by Surveillance*, which covered trials at the Delft University of Technology in Holland. The paper claimed that the technique has been used successfully in trains to ‘detect aggressive acts and violence against people and material.’

No wind or solar power here: my grandfather, WR Winstanley, rented out valve radios sets and batteries around the Hornsea area.

Back in 2009, a Delft University student thesis described the building of a ‘visual speech recogniser’ – an automatic lip reader. It undoubtedly offers some major benefits, such as machine-driven sub-titling offering accessibility for the hearing-impaired, or for spelling out conference speeches in real time on a large screen, but the ability of AI systems to ‘read our lips’ from video means that paranoid people may end up ‘cupping their hands over their mouths in public’, said the Surveillance Camera Commissioner. Unless privacy is tightly regulated, we will have to be careful what we say in public, in case cameras are listening in.

For readers interested in exploring this area of technology, desktop software can break down human speech into its core ‘phonemes’. As every primary schoolteacher knows, some 44 phonemes make up English speech (for example, see **https://youtu.be/wBuA589kfMg**). Dedicated visualisation software can map these ‘raw data’ phonemes onto computer-generated 2D or 3D characters, with every facial and lip expression – or ‘viseme’ – corresponding to a phoneme. To see this in action, start with the free lite version of the LipSync Tool from www.annosoft.com, which will ‘lipsync’ audio tracks and display phonemes that can be mapped onto visemes. (Click CTRL and + to zoom in and view the phonemes.) The unrestricted Lipsync Tool costs $500 per seat. CrazyTalk Reallusion is a long-established desktop program that claims to turn any image into a talking 2D or 3D head, and has its own audio lip-syncing engine that animates a character’s mouth. It can also handle photo-realistic images. A fifteen day trial (Windows 7+, Mac) is available from www.reallusion.com/crazytalk/ – it’s a 515MB download and the 900MB resource pack is worth having. For serious (and wealthier) desktop users, Adobe Character Animator is available as part of Creative Cloud.

Ill winds blow no good

In September’s column I mentioned the Hornsea One wind farm located off England’s east coastline in the North Sea. Presently, Hornsea claims the world’s largest offshore wind farm and is an integral part of the country’s renewable energy network. Hornsea was also the seaside resort where the author played as a young lad, when grandfather WR Winstanley toured in his van renting out valve radio sets and batteries to householders. After owning a modest radio and TV shop in Tickhill, Doncaster he went on to work with George Hollingbery and a handful of staff at a radio retailer in Hull, building what became the Comet Electrical empire. Some bench equipment inherited from WRW is still used by the writer to this day.

On 9 August, during torrential thunderstorms, parts of Britain suffered a major power cut and more than a million homes lost electricity for up to an hour. Down south, London’s St. Pancras and King’s Cross railway stations were blacked out and the rail network was thrown into turmoil as rail engineers had to manually restart stranded electric trains. The plans of thousands of stranded travellers were ruined. Initially, the media blamed the outage on storms affecting the wind turbines at Hornsea One. The National Grid incident report later showed that lightning strikes had caused a cascade of separate events in a few hundred milliseconds as the network scrambled to safeguard the supply frequency.

Lightning had struck part of the network north of London and less than a second later Hornsea One automatically ‘shed load’, dropping some 92% of its 800MW output. Simultaneously, a gas power station (Little Barford) shut down, meaning a total of 1GW or 5% of the country’s supply was lost. The supply frequency fell dramatically to 48.8Hz before the supply was gradually restored. The incident report states that almost every safety measure operated as intended, but Hornsea’s ‘controls have been fine tuned for [better] responding to abnormal events’, and we are assured the wind farm is operating robustly. I can sympathise with those engineers grappling with such unforeseen problems in real time, and I’m reminded of WR Winstanley endeavouring to keep East Yorkshire’s valve radios glowing brightly as he travelled around in his van. See you next month for more *Net Work!*

The author can be reached at:
alan@epemag.net

COCKPIT VOICE RECORDER

Accuracy better than 100 parts per billion

Part 2 – by Tim Blythman and Nicholas Vinen

Lab Quality Programmable GPS-synced FREQUENCY REFERENCE

Our new *GPS Frequency Reference* is really accurate, precise and flexible. It's also compact and easy to use, thanks to its touchscreen interface. You can set the frequencies of its three programmable outputs over a wide range (1-100MHz) and you can save preferred frequencies to a set of four presets for each output, to make switching between them quick and easy.

Last month we described the circuit of our new *GPS Frequency Reference* project and gave some details on how the software worked. We also explained its general concept and how it achieves such flexibility and accuracy in the frequencies that it can produce.

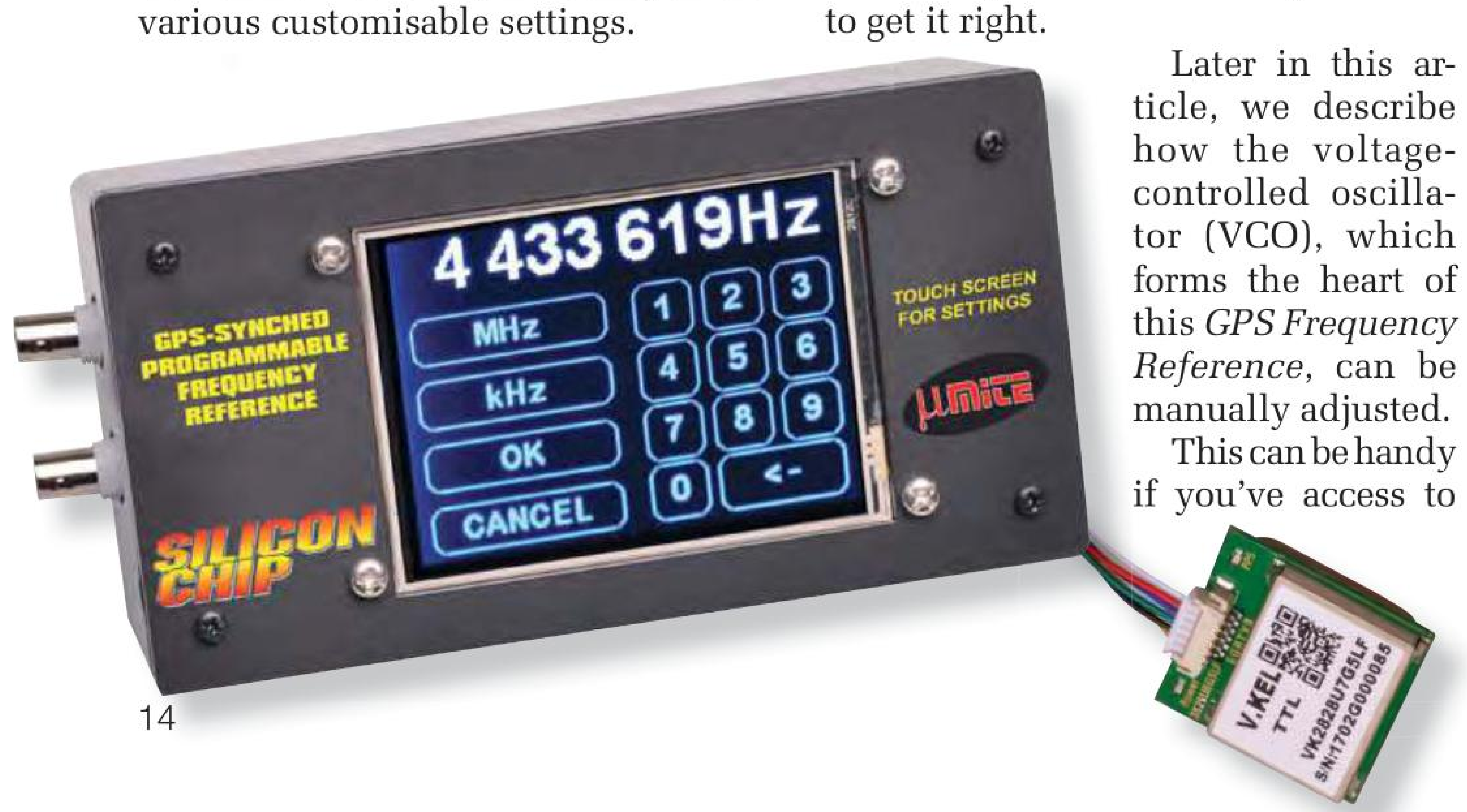
This month, we have detailed assembly instructions and further information on how to use it, including all the various customisable settings.

While the settings all have sensible defaults, allowing you to build it and start using it without any fiddling, you can tune the software parameters to suit your particular requirements.

Construction is pretty straightforward, despite the use of mainly surface-mount components. There are just one or two that are slightly tricky, but they are not that difficult, as long as you use the right tools and take your time to get it right.

Later in this article, we describe how the voltage-controlled oscillator (VCO), which forms the heart of this *GPS Frequency Reference*, can be manually adjusted.

This can be handy if you've access to a high accuracy frequency meter or can't access a GPS signal – if you're operating the unit in a basement or the middle of a steel-reinforced building.


4 433 619Hz
MHz
kHz
OK
CANCEL
1
2
3
4
5
6
7
8
9
0
<-
GPS-SYNCHED PROGRAMMABLE FREQUENCY REFERENCE
TOUCH SCREEN FOR SETTINGS
SILICON CHIP
V.KEL
TTL

Building the LCD BackPack

The first step is to build the *Micromite LCD BackPack V2*, available from **micromite.org** and described in the May 2018 issue. Do note that you cannot use the software-controlled backlight option because this uses pin 26, which we have had to use for a different purpose on the *GPS Frequency Reference* board. So you need to omit Q1 and Q2 and fit VR1 instead. The backlight brightness is then adjusted using trimpot VR1.

Once you have the parts, the assembly is pretty easy, as only about 20 components are involved and the position/polarities of most of these are printed on the PCB.

Solder the components on the PCB where shown, being careful with the orientation of the IC(s), regulator and

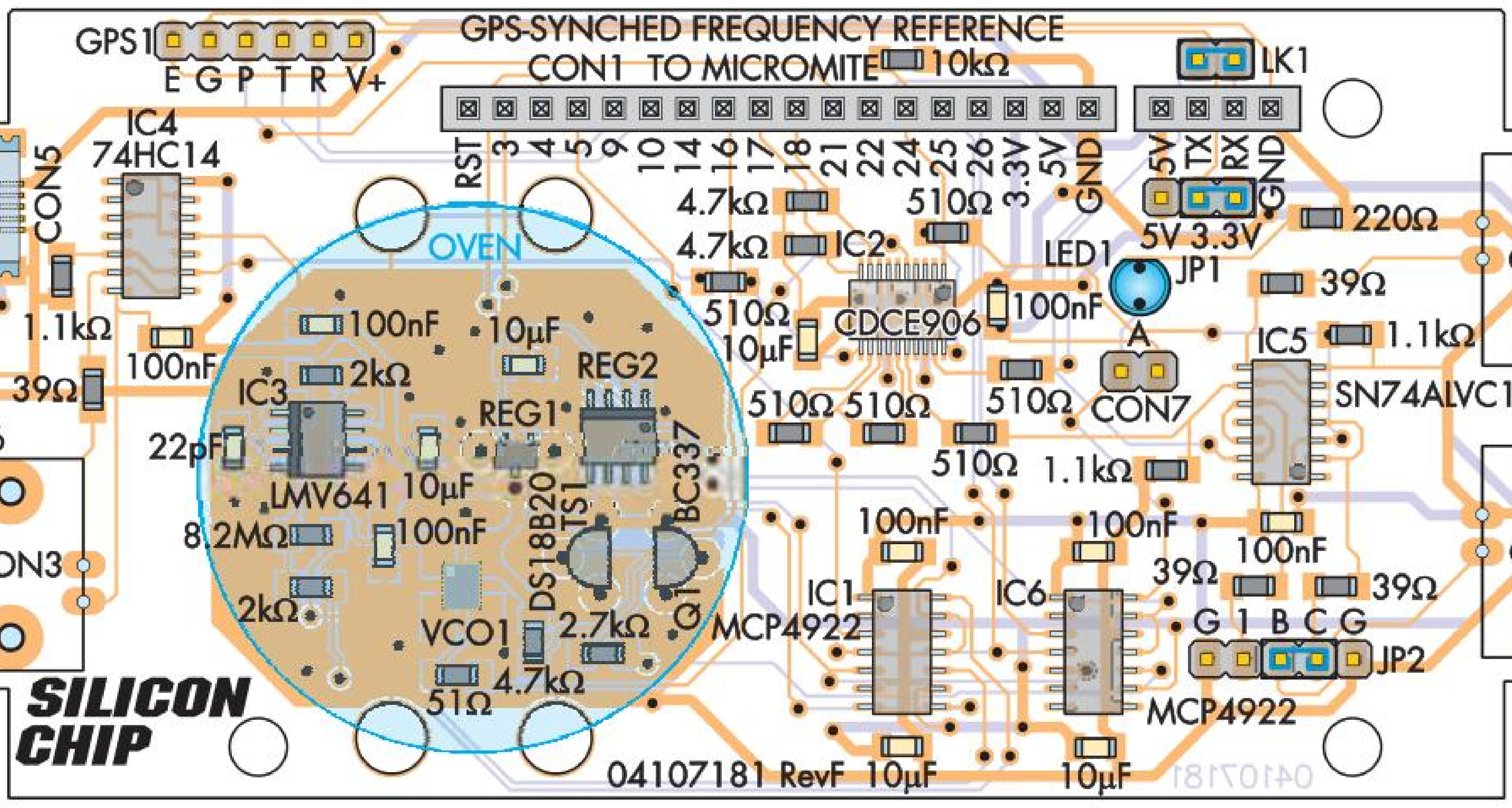
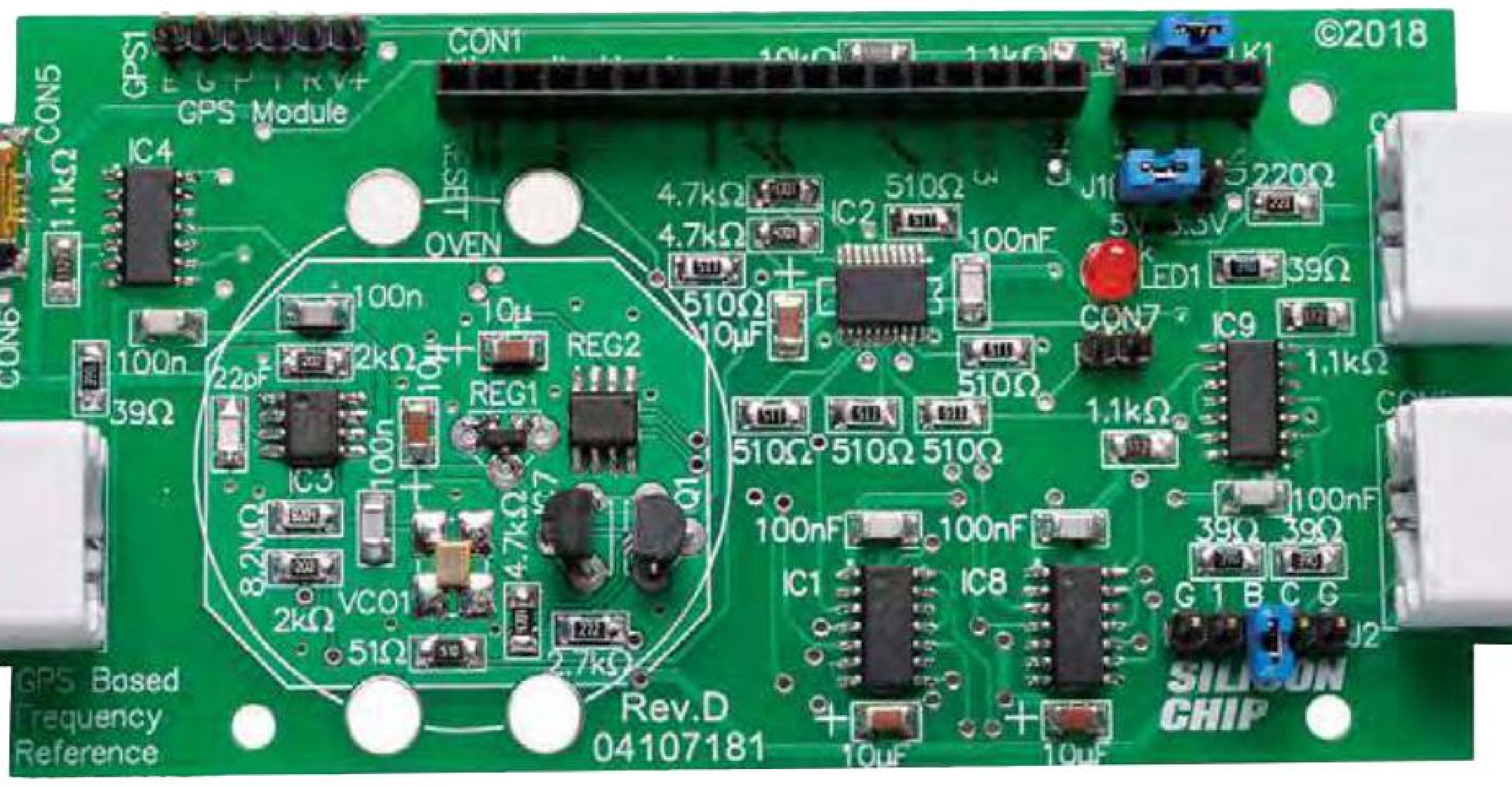

GPS1
E G P T R V+
GPS-SYNCHED FREQUENCY REFERENCE
CON1 TO MICROMITE
10kΩ
LK1
RST 3 4 5 9 10 14 16 17 18 21 22 24 25 26 3.3V 5V GND
5V TX RX GND
IC4
74HC14
CON5
OVEN
4.7kΩ
4.7kΩ
IC2
510Ω
3.3V
5V 3.3V
JP1
LED1
220Ω
CON4
1.1kΩ
100nF
100nF
10µF
2kΩ
REG2
REG1
510Ω
10µF
CDCE906
100nF
39Ω
IC5
1.1kΩ
39Ω
IC3
22pF
LMV641
10µF
510Ω 510Ω
510Ω
CON7
A
SN74ALVC14
510Ω
1.1kΩ
CON6
8.2MΩ
100nF
DS18B20
TS1
BC337
Q1
100nF
100nF
100nF
39Ω
39Ω
CON3
2kΩ
VCO1
2.7kΩ
MCP4922
IC1
IC6
G 1 B C G
JP2
CON2
51Ω
4.7kΩ
MCP4922
SILICON CHIP
04107181 RevF
10µF
10µF

Fig.3: use this diagram as a guide when building the *Frequency Reference*. The orientations of IC1-IC6, LED1, REG2 and TS1 are critical, so take care to fit them the right way around, as shown. You only need to fit one of CON5 or CON6, not both. Note the approximate location of the bottle cap or similar cylinder which encloses the oven section of the board.



LED. The capacitors supplied in the kit will not be polarised types.

Start assembly with the lowest profile components first and work your way up to the taller ones.

Note that the 18-pin and 4-pin headers are mounted on the back of the board and these should be fitted last.

You can then plug the screen into the provided header socket and attach it to the *BackPack* board using some short machine screws and tapped spacers.

Trim the solder joints on the top of the LCD with some sharp sidecutters so they do not interfere with the lid when fitted later.

PCB assembly

Next, we'll assemble the *GPS Frequency Reference* PCB. Use the overlay diagram (Fig.3) as a guide to fitting the components.

Before starting, check that you have all the components needed. If you have a kit of parts, don't pull them all out yet as some are hard to distinguish from others, especially those which have no markings (eg, ceramic capacitors).

We will refer to the orientation as though the board is sitting as shown in Fig.3, with the single BNC socket (CON3) at the left, and the two BNC sockets (CON2 and CON4) to the right.

This orientation is convenient since most of the labels are right-side-up.

It's easiest to start with the fine-pitch ICs first, as these are more difficult to solder once surrounding components have been fitted. So start by fitting IC2, the CDCE906 PLL IC.

This part is only available in an SSOP SMD package with a 0.65mm lead pitch. It requires the most care to solder but it is not too difficult to do by hand if you are careful.

The other components are much larger and have leads spaced further apart so, after this chip, it's downhill all the way. Start by applying a thin layer of flux paste to the pads for IC2, then line up the chip with the pads, ensuring the pin 1 marking is to upper right.

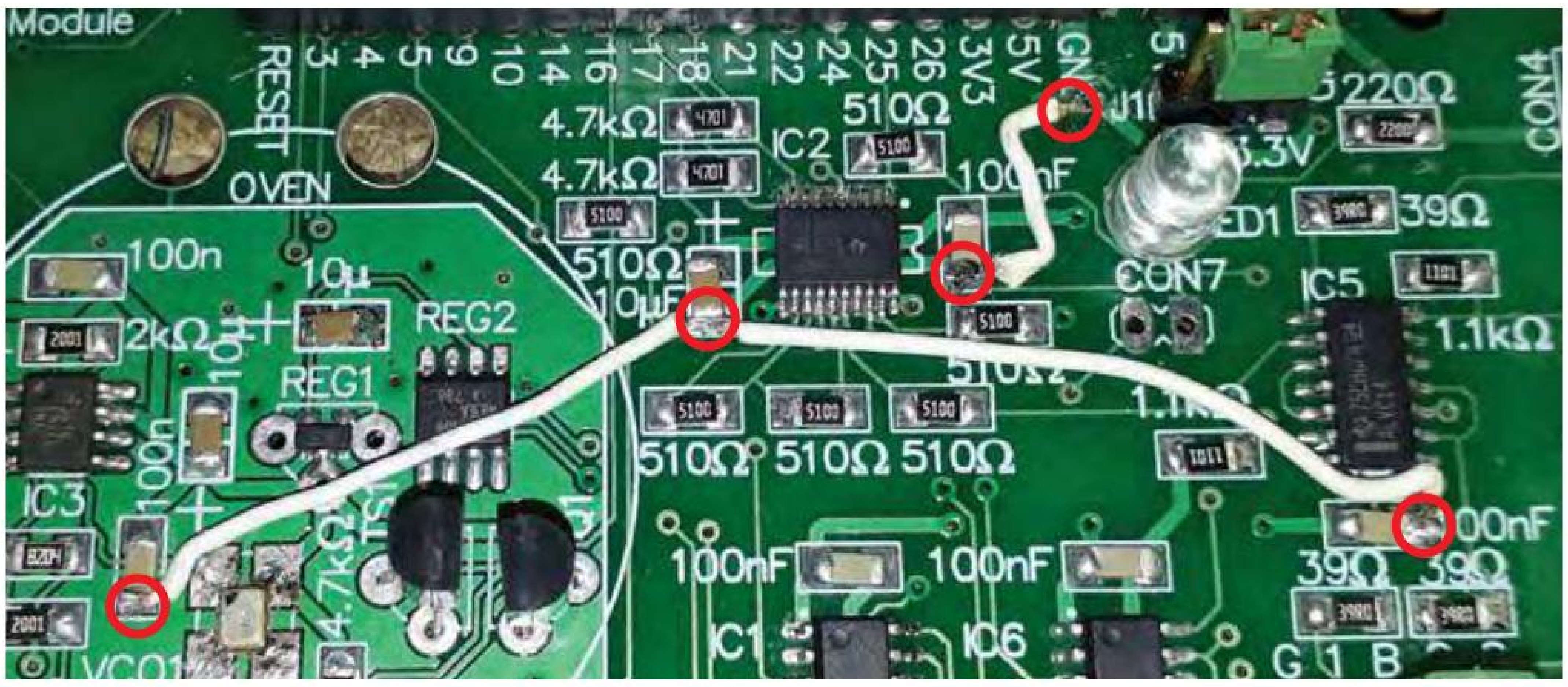
Using a fine-tipped soldering iron, tack solder one of the corner pins down and check that all the IC pins line up in the centre of the PCB pads and that the IC is flat on the PCB.

If you are happy with the location of IC2, carefully solder each pin. If you have used flux paste as recommended, simply touching the iron to the pin and pad at the same time should cause a small amount of solder to flow onto both.

You will need to add a bit of extra solder to the iron from time to time. At this stage, if there are solder bridges between pins, don't worry about them.

The important thing is to make sure that all the pins are soldered properly. Patience, and keeping the tip clean of impurities like dark oxides will make this process easier. Once all the other pins have been done, go back and retouch the first pin. If you have some solder bridges (which are almost inevitable), apply some fresh flux and use solder braid (wick) to remove the excess.

Check deep between the pins, as a single hidden bridge is enough to cause trouble. We've found taking a photo



Note: the stability of the reference may be improved by reducing the impedance of ground tracks on the PCB. This can be done by soldering a wire from a ground pad near VCO1 to the ground end of IC2's bypass capacitor, then another wire from there to the ground pad of IC5's bypass capacitor and also to the via near the GND terminal of CON1.

Here are two views of the completed PCB, along with our highly technical purpose-designed oven (in situ on the right). If it looks just like a milk bottle cap, well …

with a smartphone camera can allow us to zoom in and see bridges and other defects that aren't immediately obvious to the naked eye.

Next on the list are the USB sockets, which can be a bit fiddly, but fortunately, you only need to install one of them.

We chose the mini-USB socket beause it is slightly larger and easier to handle, but the micro-USB socket is now the more common type in use (especially on phones), so you can fit that if you prefer.

Again, start by putting a little flux paste on the pads. Place the socket on the PCB and its pegs should drop into the provided holes in the PCB, making alignment easier. Solder the large mechanical pads first, making sure the socket is flat and flush with the board.

Now carefully apply a little solder to each of the small leads to lock them in place. We only need the pins at either end for power, but it's probably a good idea to solder them all anyway.

Be sure to check that the USB data pins are not bridged to the power pins, as this may cause problems if the *GPS Frequency Reference* is powered from the USB port on a computer.

You can now fit IC1 and IC4-IC6, using a similar process as for IC2. These are considerably larger and easier to handle.

Check that the pin 1 markings are correct. IC1, IC4 and IC6 have their dot facing upwards, while IC5 has its dot facing downwards.

If there is no dot, you might find a bar on one end of the IC, or even a bevel along one edge. In each case, pin 1 is close to the dot/bar/bevelled edge.

Next on the list are REG1, REG2 and IC3, which are all within the oven outline. This is marked with a circle and there is also a corresponding copper pour on the PCB.

While this should not present any difficulties, you might find that the extra mass of copper pulls heat away from the iron, so you may need to turn its temperature up slightly to compensate.

If you are using the SOT-23 version of REG1 then it should be soldered first, as it is quite low. It will only fit one way, so tack one leg in place, check the alignment and then solder the other two leads and touch up the first pin.

If you are using the TO-92 version of REG1, you can fit it later, once all the SMDs are in place. IC3's pin 1 goes towards the upper left corner while REG2's pin 1 goes to the top right.

Solder these components using the same technique as the other ICs. Now is a good time to solder VCO1.

The pin 1 marking on this module is one of the smallest we have seen. If you cannot find it, then rotate your PCB so that CON3 is at the bottom. Then place the VCO on the board so that the writing on it is right-way-up. You might now see the small marking at the bottom left, matching the dot on the PCB.

For smaller components like this, just adding a small amount of solder to one pad before placing the component means that you don't have to apply solder while trying to position the component.

Use tweezers to hold the component flat and aligned while adjusting its position, then when you are happy, solder the other leads in place.

The VCO's pads are much larger than necessary, to make it easier for you to get the iron in contact with them despite the tiny size of the device. Ensure VCO1 is symmetrical about the pads so that each one makes good contact.

Carefully apply more solder if necessary, but avoid getting any near the top of the VCO, as it may stick to the metal can and cause problems later.

Passive components

The passives should be fitted next. The capacitors are not usually marked, so only take them out of the package one value at a time. Fortunately, they are not polarised. Fit them where shown in Fig.3.

Follow with the resistors. There are several different values, but fortunately they are marked with codes indicating their values if you get them mixed up.

Fit them in the same manner as the capacitors and again, refer to the overlay diagram to see which goes where.

The final items are three 1.1kΩ resistors. You may have noticed that we had four 1.1kΩ resistors in our original parts list, but there are only three on the board.

We found that the resistor on IC2's Y4 output was limiting the swing on the 40MHz signal going back to the Micromite, so its frequency wasn't being measured accurately. Thus, we removed this resistor from the final design.

Through-hole components

Now is a good time to fit Q1, TS1, LED1 and (if you are using the through-hole version) REG1. Ensure LED1 is inserted with its longer anode lead through the pad marked 'A' on the PCB. The orientation of the TO-92 package devices is shown in Fig.3, but you may need to bend their leads out (eg, using small pliers) to fit the pads provided.

Now you can solder the headers in place. This includes CON1, GPS1, LK1, JP1 and JP2. These are all fitted on the same side of the board as the other components and, except, for CON1, they are standard pin headers.

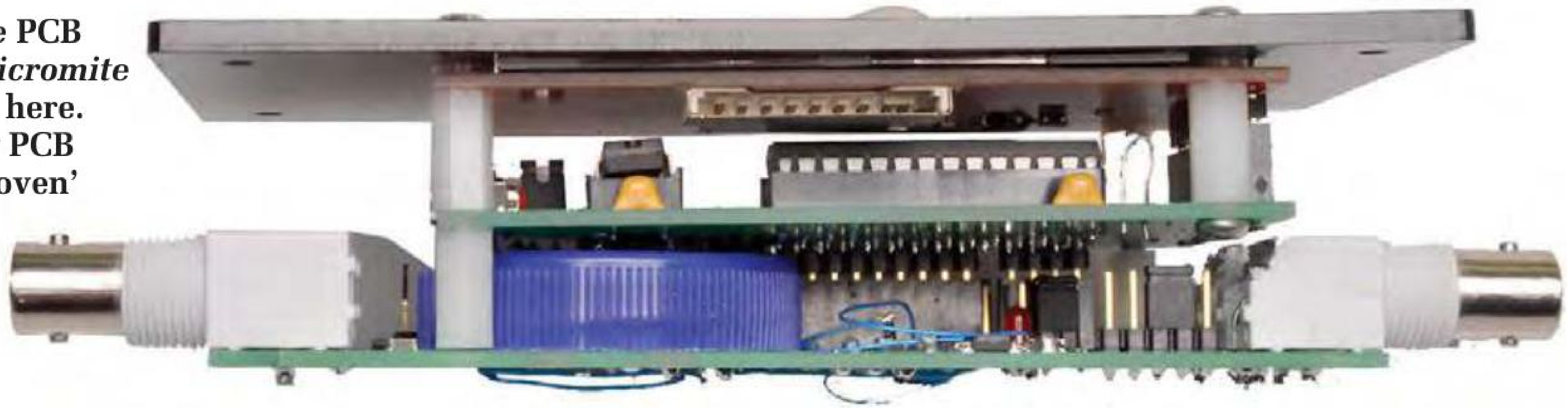
CON1 consists of two female header sockets, one with 18 pins and one with four pins. You can cut these down from longer sockets if necessary.

When fitting the GPS header, be careful to ensure it is perfectly vertical; otherwise, it may be difficult to plug the GPS connecting wires into it later.

To make sure they will fit, it's best to plug the *BackPack* into the *GPS Frequency Reference* board after soldering

SMD Resistor Codes

☐	1	8.2MΩ	825	*or*	8204
☐	1	10kΩ	103	*or*	1002
☐	3	4.7kΩ	472	*or*	4701
☐	1	2.7kΩ	272	*or*	2701
☐	2	2.0kΩ	202	*or*	2001
☐	1	1.1kΩ	112	*or*	1101
☐	6	510Ω	511	*or*	5100
☐	1	220Ω	221	*or*	2200
☐	1	51Ω	511	*or*	51R
☐	4	39Ω	390	*or*	39R



The *Frequency Reference* PCB 'hangs' underneath the *Micromite BackPack* PCB, as shown here. The *Micromite BackPack* PCB also holds the bottle cap 'oven' in place. If you mount it differently, the cap will need securing to the PCB via the holes provided.

one or two pins on GPS1, so that you can check that the header clears the board above.

To help line the CON1 sockets up correctly, you can plug them into the corresponding headers on the *Micromite LCD BackPack* first, and then insert them into the pads on this PCB and solder the pins in place. The final items to fit are the three BNC sockets, CON2-CON4. The large posts require a decent amount of solder to hold them in place (and heat to make those solder joints).

If you're building the unit into a larger box than specified, you could run some shielded cable out to chassis-mounted sockets.

Setting up the BackPack

If you haven't used a PIC pre-programmed with the software for this project, you will need to set up the LCD screen and touchscreen. You can do this by connecting a USB/serial adaptor to the 4-pin header and plug it into your computer, then open up a terminal program, select the correct COM port and set the baud rate to 38,400.

Reset the Micromite and you should receive a greeting banner in the console. If you don't, check the serial wiring, COM port, baud rate, power supply and that you have assembled the PCB correctly.

Assuming you get the greeting, you can set up the display and touch controller with the following commands:

```
OPTION LCDPANEL
    ILI9341, L, 2, 23, 6
OPTION TOUCH 7, 15
GUI CALIBRATE
```

You then need to use a sharp object (but not too sharp!) like a toothpick to press on the middle of the targets which appear on the screen. Once you've done that, you're ready to load the BASIC software for this project.

Loading the software

Now that the two PCBs have been assembled, plug them together but leave all the jumpers off for now. The next step is to load the BASIC software onto the microcontroller.

If you have a PIC chip in your *BackPack* that was pre-programmed with the *GPS Frequency Reference* software then you can skip right to the testing stage.

We suggest that you then use the MMEdit software to upload the BASIC program, and the following instructions assume you will be using this method. If you are familiar with using the Microbridge to upload HEX files directly to the chip then you can do that instead.

Open MMEdit and load the BASIC file for this project, which is available from the November 2019 page on the *PE* website. Connect the Micromite to your PC via the USB socket on the *BackPack* itself (not the one on the *GPS Frequency Reference* PCB).

Under the Connect menu, select New and find your Serial port number, then select it. Set the baud rate to the rate your Micromite is set up for (the default is 38,400).

Under the Advanced menu, ensure that the 'Auto Crunch on Load' option is selected. This is necessary as the program will not fit into the Flash memory without being 'crunched'. Press the button to upload the code and when it finishes, type:

```
OPTION AUTORUN ON
```

into the console which appears and press Enter. This sets the program to run next time the unit is powered up.

Wiring up the GPS module

There is not much spare room in the specified enclosure for the GPS module and anyway, you will probably get better results by mounting it externally, as we have on our prototype.

Alternatively, you could use a module with an external antenna connector and mount a socket just above the USB power socket on the case.

Since the GPS header (GPS1) is so close to the *BackPack* board above it, we recommend that you use slim DuPont-style headers to make the connections. The pins are labelled as follows:

V+ (module power supply),
R (goes to Rx/RxD on the module),
T (goes to Tx/TxD on the module),
P (goes to 1PPS output on the module),
G (GND) and
E (enable – connected to V+).

If you are not using the recommended module then it may not have an enable pin, or it may require a different voltage. You will need to use a module with a 1PPS output and TTL serial interface.

Testing

Close the console and unplug the USB cable from the Micromite. Insert a jumper on the LK1 header. This will connect the VCO output to the Micromite's pin 12, and also ensure that the console does not start up and interfere.

Plug a powered USB cable into the USB socket on the *GPS Frequency Reference* PCB and observe LED1. It should fade on and off for a few seconds.

At this stage, everything should be working and the splash screen should now be shown. To follow the status of the startup, press the 'Status' button.

The six lines at the bottom of the Status screen are the important ones to watch, as the top lines are mostly information taken from the GPS module's NMEA data.

You may not see all items go to 'OK' in the startup page straight away, particularly the GPS-related items, as the GPS module usually takes some time to achieve a satellite fix.

If you are using the VK2828 GPS module, you will know when it has a fix – its green LED will start flashing.

The 'Temp Sensor' line should read 'OK' and the temperature should be rising or near the setpoint. That means the oven is working correctly.

If 'Temp Sensor' shows 'Not ready' then TS1 is not wired correctly. If 'Temp Sensor' shows 'OK' but the temperature is not rising, there is a problem with Q1, the 2.7kΩ resistor or DAC IC6. If the LED was fading initially then the DAC is probably working.

There are three lines which indicate the status of the GPS module. The first one to check is 'GPS Receiver'. If that does not show 'OK' then no data is being received and you should check the GPS module's wiring.

The 'GPS 1PPS' and 'GPS Locked' status lines will typically be the last ones to show 'OK', as they depend on the GPS module having a good satellite fix. If you are testing indoors, you may find they flick between 'OK' and 'Not ready'.

The ‘PLL unit’ and ‘VCO output’ lines are only updated at startup, so will not change if left for a while. If ‘PLL unit’ does not show ‘OK’ then the Micromite cannot communicate with IC2. This may be due to problems with the I^2C bus.

‘VCO output’ shows OK when the Micromite detects a ~40MHz signal. That means that the PLL and VCO are working to some extent. If there is no ‘VCO output’ then check that the VCO chip is soldered to the PCB correctly.

As the 40MHz signal to the Micromite is also fed through the PLL (IC2), you should confirm that there aren’t any problems with IC2 as well; for example, solder bridges between pins or bad solder joints.

Another test that you may like to do if you have an oscilloscope or frequency counter is to check that there is an output from each of the BNC sockets (or the JP2 header).

Assuming that JP2 is set to the ‘BC’ position, all of CON2, CON3 and CON4 should be producing a 40MHz signal. If this is the case, then it is time to complete assembly.

Finishing the oven

While you would have seen the temperature of the oven increasing on the status page, and the unit is effectively functional, we can add some insulation to the oven to improve its ability to hold heat.

This helps to ensure that the temperature inside the oven is uniform, so that the temperature measured by TS1 more closely reflects the temperature of the other components inside the oven.

We’ve sized the oven to be roughly the same diameter as a bottle cap from a two-litre milk bottle. We’ve found that some of them also have a foam insert which provides extra insulation.

The height of our cap was precisely 12mm, which matches the tapped spacers between the two boards. Unfortunately, due to the components on the *Micromite BackPack* PCB, the available space is reduced slightly, and the rim of the bottle cap will probably need to be trimmed.

It’s a good idea to give the lid a thorough clean with soap and hot water to ensure there is no milk residue. We can imagine nothing worse than a *GPS Frequency Reference* that smells like mouldy cheese!

If the lid is a snug fit between the *Micromite BackPack* PCB and the *GPS Frequency Reference* PCB, it can simply be sandwiched in place. Otherwise, holes are provided on the PCB for cable ties to hold it in place.

Alternatively, you could use a small amount of neutral-cure silicone sealant around the rim to seal it and stop it from moving around. The underside (ie, non-component side) of the PCB should ideally be insulated as well.

You could either use a foam insert from another milk bottle (held in place by the same cable ties) or merely apply some foam-backed double sided tape to the back of the PCB.

PCB jumper settings

The jumper on JP1 selects whether the GPS module receives 3.3V or 5V. Most modules will run off 3.3V, including the VK2828U7G5LF, but if you are not sure, check the module’s data sheet.

Fit a jumper shunt between the pins labelled B and C on JP2 if you want a programmable frequency on CON2. Alternatively, fit the shunt between the pins labelled 1 and B for a (disciplined) 1PPS (1Hz) output from CON2.

The pins labelled ‘G’ are connected to ground, so you can run a shielded cable from the pairs of pins at either end to a chassis connector if you decide to make both of these signals available externally.

A shunt is placed on LK1 for normal operation, but this prevents programming the Micromite chip, so remove it if you need to reprogram the chip. CON7 is for debugging the software, so you can safely ignore it unless you plan to modify the software.

Putting it all together

Now power down the *GPS Frequency Reference* and detach the *Reference* PCB from the *BackPack*. If the *BackPack* display is attached by screws, remove them to allow the front panel to be fitted.

Assuming everything is apart, start by attaching the LCD to the laser-cut acrylic UB3 lid panel, using M3 machine screws at the front and tapped spacers at the back.

Insert 1mm nylon washers between the lid and LCD to provide clearance for the solder joints.

Use 20mm-long machine screws on the bottom left, bottom right and top right holes. This is with the touch panel oriented so that its flex connector is on the right, along with the 14-pin header.

For the top-left machine screw, use one of the shorter ones initially fitted to the *LCD BackPack*, again with a tapped spacer on the back. This is necessary because a fourth long screw would interfere with the GPS header.

The *Micromite BackPack* PCB can now be inserted over the three long machine screws shafts and can be loosely secured with a short machine screw into the single tapped spacer.

Now feed the tapped spacers over the three remaining screw shafts. Ensure everything is tight and lines up. In particular, check that the LCD’s screen is flush with (or slightly behind) the lid panel.

Finally, attach the *GPS Frequency Reference* PCB to the back of the *Micromite BackPack* PCB using the three remaining short machine screws from the original *BackPack* kit.

Putting it in the box

The enclosure specified is a standard UB3 Jiffy box. You will need to make cutouts at two ends for the BNC and USB sockets; see Fig.4 for details. You only need to make one of the cutouts for the mini-USB and micro-USB socket, depending on what you fitted to the board.

We used a stepped drill to make the BNC socket holes, although you could use a standard drill and then enlarge them to size with a tapered reamer.

We made the vertical slot for CON3 using a hacksaw, cutting straight down from the top of the side of the box.

The holes for the USB sockets can be started with a small drill bit and completed with a file.

If you are feeling lazy, or don’t enjoy cutting square holes, you could make (slightly larger) round holes for the USB sockets.

You may find that you have to make the hole larger than shown in the diagrams if the shroud on your USB plug is unusually large.

The final step is to carefully thread CON2 and CON4 into the holes in the right-hand side of the case and then lower CON3 down into its slot. Check that all the holes line up and that a USB cable will plug in. Then attach the acrylic lid to the Jiffy box using the supplied screws (or longer ones, if the ones that came with your box are too short) and fit the nuts and washers to the BNC sockets.

Using it

As you are reading the following instructions, you may wish to refer back to the first article on this project in last month’s issue, as it included images showing many of the screens described below.

Once power is applied via the USB socket, the start screen will show for three seconds, after which the main screen appears. Press the STATUS button to check that everything is working as expected. The BASIC program is quite busy processing data, so sometimes it is necessary to press on the buttons for more than a brief ‘tap’.

The Temperature line shows the current oven temperature and setpoint, followed by the oven heat controller DAC output, where zero is off and 4095 is full power.

After the unit has tuned itself and the oven temperature has reached its set point, it will provide a high degree of accuracy.

If your workbench area typically gets above the 35°C we have set for the oven, you may need to make the setpoint a bit higher, so that the oven has a consistent temperature in all conditions.

See the Settings section below for details on how to do that.

The 'GPS 1PPS' and 'GPS Locked' status lines need to show 'OK' before oscillator disciplining occurs, but once the unit has got past the start screen, it is effectively operational, although it will not yet be operating with full accuracy or precision.

The tuning algorithm waits until it has received 1000 1Hz pulses with GPS lock, then calculates the average oscillator frequency (as seen on the top line) and adjusts the VCO control voltage to bring it closer to 40MHz.

Given that 1000 pulses take about 17 minutes and it takes some time for the GPS receiver to get a satellite fix, it should begin tuning itself within about half an hour of power-on.

You can explore the features of the unit before it has fully tuned itself; the initial tolerance on the VCO is 2ppm, which makes it a useful tool straight away.

Pressing the 'BACK' button to go back to the main menu, you can jump straight into any of the adjustment pages for CON2-CON4 to set their output frequencies.

Note that these labels are adjacent to their respective BNC sockets, which helps you to remember which is which.

Once you've entered one of the output setup screens, the 'SEEK F' button allows a frequency to be entered on a keypad and the unit will find the nearest frequency that it can synthesise to what you enter.

It will show the frequency, the various PLL dividers and even the PLL's internal frequency to allow you to decide if that particular combination is suitable.

Pressing 'OK' will then update the PLL parameters to those shown and the new frequency will be immediately available from that output socket. Press 'CANCEL' to go back to the output setup screen without changing the output frequency.

Manually setting up the PLLs

The 'ADVANCED' page permits manual selection of the N, M and P dividers in the PLL, except for CON4, where only the P value can be changed; the N and M values are fixed because this PLL is shared with the output that provides VCO feedback to the Micromite.

As explained last month, the incoming 40MHz signal is multiplied by N and divided by M to give the PLL frequency and then divided by P to give the output frequency.

While the PLL is supposed to operate between 80MHz and 300MHz, we found that it worked outside this range (perhaps with more jitter). The PLL frequency is displayed near the top of the page, and if it is out of range, it is displayed in yellow. In this case, you should verify that the output frequency is accurate and stable.

If the resultant output frequency is above 99999999Hz, it is displayed in red. Although such frequencies can be set, they appear to be very unstable and may cause the PLL to stop functioning.

In any case, the output buffers will not work well above 100MHz, so we really do not recommend that you use such frequencies.

The only conditions that are enforced when you enter the PLL configuration manually are that N is between 1 and 4095, M is between 1 and 511, P is between 1 and 127 and that N is greater than M.

Like with the other configuration screen, once you have set parameters that you are happy with, press the 'OK' button to update the output frequency or the 'CANCEL' button to return to the previous screen without making any changes.

Using frequency presets

The CON2-CON4 setup pages also show four preset frequencies. They are initially 80MHz, 40MHz, 20MHz and 10MHz (all using a PLL frequency of 160MHz).

The frequency of the output can be changed to any of the presets by pressing that button briefly. Or, to change one of the presets, set the output to the desired frequency and then hold

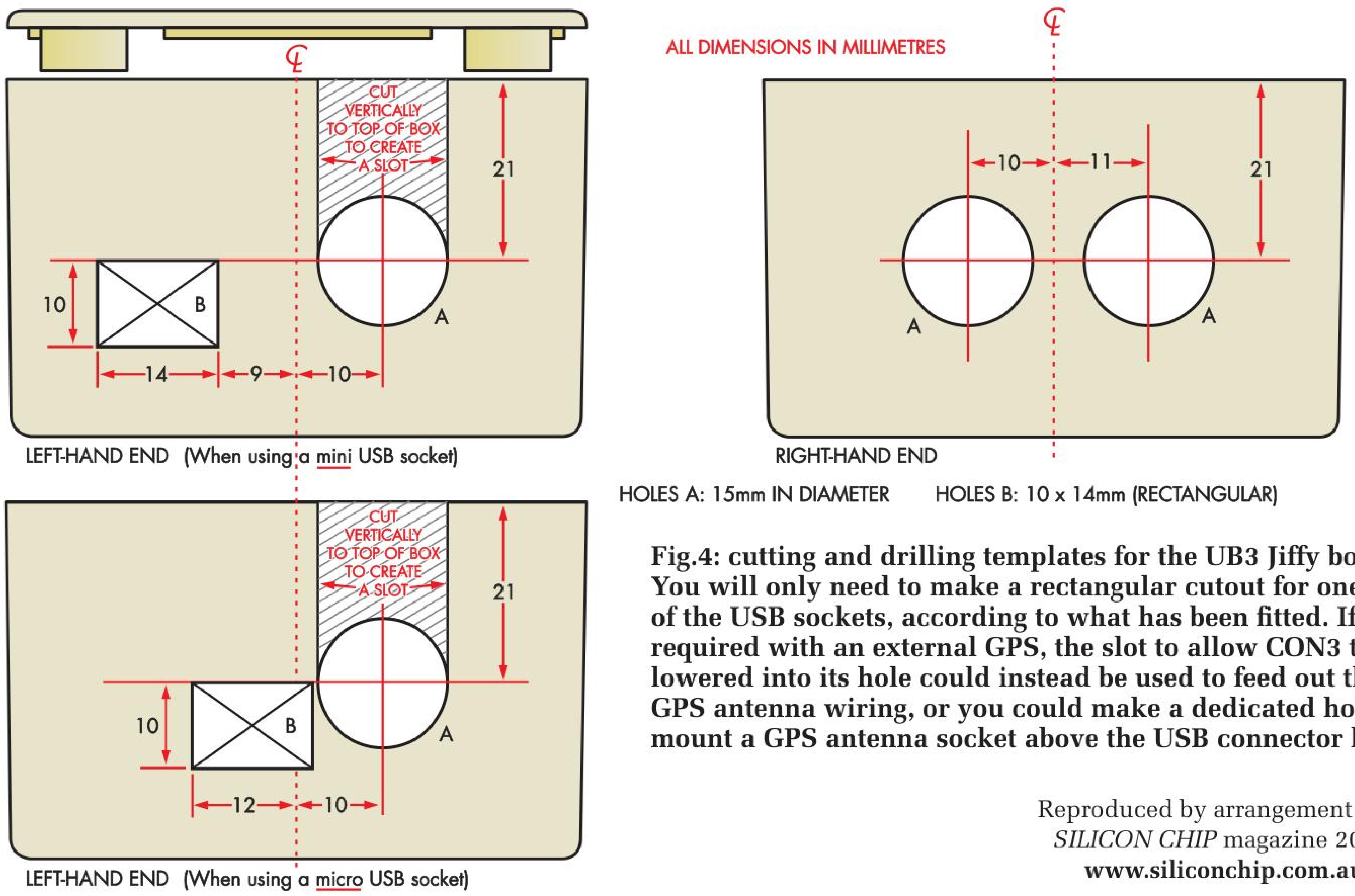
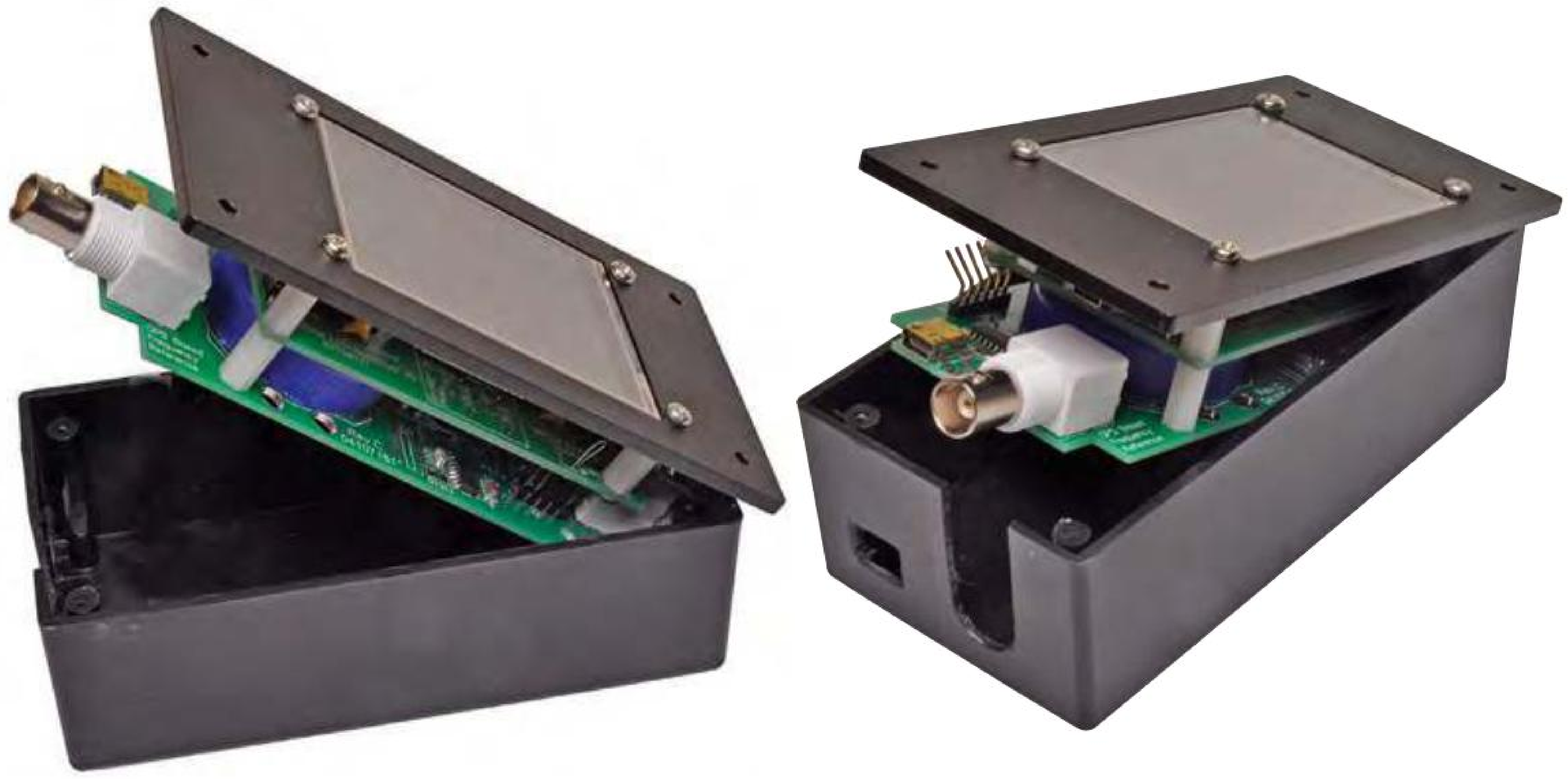

ALL DIMENSIONS IN MILLIMETRES
℄
CUT VERTICALLY TO TOP OF BOX TO CREATE A SLOT
21
10
B
A
14
9
10
LEFT-HAND END (When using a mini USB socket)
℄
10
11
21
A
A
RIGHT-HAND END
HOLES A: 15mm IN DIAMETER
HOLES B: 10 x 14mm (RECTANGULAR)
CUT VERTICALLY TO TOP OF BOX TO CREATE A SLOT
21
10
B
A
12
10
LEFT-HAND END (When using a micro USB socket)

Fig.4: cutting and drilling templates for the UB3 Jiffy box. You will only need to make a rectangular cutout for one of the USB sockets, according to what has been fitted. If required with an external GPS, the slot to allow CON3 to be lowered into its hole could instead be used to feed out the GPS antenna wiring, or you could make a dedicated hole or mount a GPS antenna socket above the USB connector hole.

Reproduced by arrangement with
SILICON CHIP magazine 2019.
www.siliconchip.com.au



Fitting the assembly to the UB3 case is a little tricky – but it can be done! After drilling/filing the required case holes, you need to introduce the boards to the holes for CON2 and CON4 at quite a steep angle, as shown here. If your holes are accurately drilled, the board should slip into place quite easily and then it's simply a matter of lowering it all into place so that CON3 and the USB socket mate with their holes on the left-hand end. There is no need for any screws holding the board from underneath – the screws which hold the front panel in place hold the whole assembly snug and secure.

down the preset button for more than one second.

There is also the option to copy presets between the outputs by using the 'PRESETS' page, which can be accessed via the 'SETTINGS' button on the main page. The preset page has two buttons at the top to allow you to scroll between the various connector presets and output value settings. Their current values are displayed below.

Further down, there is a 'COPY' button and a 'PASTE' button, followed by the current 'clipboard' values. Pressing the 'COPY' button copies the currently selected preset or output value to the clipboard and pressing 'PASTE' copies the clipboard value back to the preset or output value. A 'BACK' button is provided to return to the main page.

The software will give an error message if you try to copy any setting to CON4 which is not compatible; ie, it does not have N=4 and M=1.

Additional settings

On the 'SETTINGS' page, there are also options to adjust the oven 'TEMPERATURE' control loop and the 'VCO TRIM' settings. Under the 'TEMPERATURE' menu, there are options for Setpoint, Gain and Offset.

The Setpoint is the target temperature of the oven, and as we mentioned earlier, it should be higher than the highest expected ambient temperature where the unit is being used. The default is 35°C, which is suitable either for colder regions or buildings with air conditioning.

The Gain and Offset values are used to change the behaviour of the control loop. It uses simple proportional control; the default values of 1000 for Gain and 3000 for Offset work well. It's unlikely that you would need to change them unless your transistor Q1 has a wildly different gain from the components that we used in our prototypes.

Both values are in DAC step units (out of 4095) with Offset being the DAC output level when the target temperature is reached and Gain being the change in DAC output level for a 1°C error.

If you find that the oven temperature is oscillating wildly, the Gain value should be reduced. A small amount of drift (under 1°C) is to be expected and is not a cause for concern.

If you find that the oven temperature is consistently too high or too low, adjust the Offset value. Allow the unit to settle for about 10 minutes, then check the current DAC output (the number in brackets on the STATUS page) and enter this value as the Offset. You may need to repeat this a few times to get an ideal value.

If you change these values, press the 'SAVE' button to store the changes (shown in yellow) or the 'BACK' button to go back to the settings screen without making any changes.

Adjusting the VCO control loop

The final settings page is for adjusting the VCO control parameters, which include a 'Gain' value, a 'C Value' (control value) and the 'Update s'.

The 'Update s' value is the number of 1PPS pulses that are counted before an adjustment is made to the VCO. The default is 1000, but this can be extended to provide further precision, as more 1PPS pulses will be sampled.

The C Value is the current VCO control DAC value (0-16,777,215). This is the value that is changed by the disciplining routine after the correct number of 1PPS pulses has been received.

As such, you should see the value change as this occurs. The default value is chosen to be at the midpoint of the VCO's pulling range.

The Gain value sets the number of DAC steps by which the C Value is changed per Hertz of error, and has been calculated as follows.

The VCO has a pulling range of 0.5 to 2.5V, corresponding to a frequency change of 10ppm (from –5ppm to +5ppm around nominal). The DAC's voltage reference is nominally 2.5V, so the span of the 2.0V pulling range corresponds to 13,421,772 DAC steps.

With a 40MHz nominal frequency, the 10ppm range of variation corresponds to 400Hz. Dividing 13,421,772 by 400 gives 33,554 DAC steps per Hertz, which is our calculated Gain value. Another way to look at this is that each DAC step corresponds to a change of around 30μHz in the VCO output, which gives very fine control.

This is all designed to ensure that the *GPS Frequency Reference* converges as quickly as possible on the first round of disciplining; given that this process is repeated, the unit is also able to adjust for drift and other factors automatically.

Once again, use the 'SAVE' button to commit any changes to Flash memory.

Manual VCO calibration

If you have an accurate frequency counter, you can use this to adjust the VCO manually, using the calculations above. If you want to disable automatic adjustment, you can either remove the GPS module or set the VCO 'Gain' value to zero. The 'C Value' will then remain constant.

To manually trim the VCO, allow the oven temperature to stabilise and set one of the outputs to 40MHz (they are set to this by default in the initial firmware settings). Check the frequency using a precision frequency meter and note the offset in Hertz.

Take this offset, and multiply it by the 33,554 value we calculated earlier, and add it (if the current frequency is too low) or subtract it (if the frequency is too high) from the current 'C Value'. If there is a small residual error, you can repeat the adjustment to tweak it further.

Conclusion

That completes the construction and set-up of the *GPS Frequency Reference*.

We are sure that you will find it useful; we certainly plan to make good use of our prototype.

Win a Microchip SAM R30M Xplained Pro Evaluation Kit

Practical Electronics is offering its readers the chance to win a Microchip SAM R30M Xplained Pro Evaluation Kit (AC164159).

The SAM R30M Xplained Pro is a hardware platform designed to evaluate the SAMR30M18A module. It is supported by the Atmel Studio, an integrated development platform that provides predefined application examples. This kit provides easy access to various features of the SAMR30M18A module and offers additional peripherals to extend the features of the board and ease the development of custom designs.

The SAM R30M Xplained Pro is the industry's smallest IEEE 802.15.4-compliant module that combines an ultra-low-power microcontroller with a sub-GHz radio, accelerating time to market and providing long-lasting battery life in wireless-networked sensors. At half the size of the next smallest module on the market, the SAM R30 module meets the needs of space-conscious designs such as home automation sensors and controls.

Based on IEEE 802.15.4, the SAM R30 module supports proprietary networks that can be easily customised and configured; ideal for applications where interoperability is not desired due to their inherent vulnerability to remote attacks, such as alarm systems, building automation, smart cities and industrial sensor networks. A key advantage of an IEEE 802.15.4-based network is that member devices can sleep for extended periods of time and remain part of the network. The SAM R30 module features ultra-low-power sleep modes, with wake from general-purpose input/output (GPIO) or its built-in real-time clock (RTC) while consuming approximately 800nA. Devices can sleep for years, only waking as needed to transmit data.

Developers can begin prototyping with the SAM R30M Xplained Pro, supported by the Atmel ICE Debugger/Programmer (ATMEL-ICE) and the easy-to-use Atmel Studio 7 Integrated Development Environment, with demo code examples supporting each network configuration.

Worth
$38.00
(approx £31.25)
each

Free-to-enter competition

Microchip SAM R30M Xplained Pro Evaluation Kit

How to enter

For your chance to win a Microchip SAM R30M Xplained Pro Evaluation Kit, enter your details in the online entry form at:

http://page.microchip.com/PE-ATSAMR30M-XPRO-Board.html

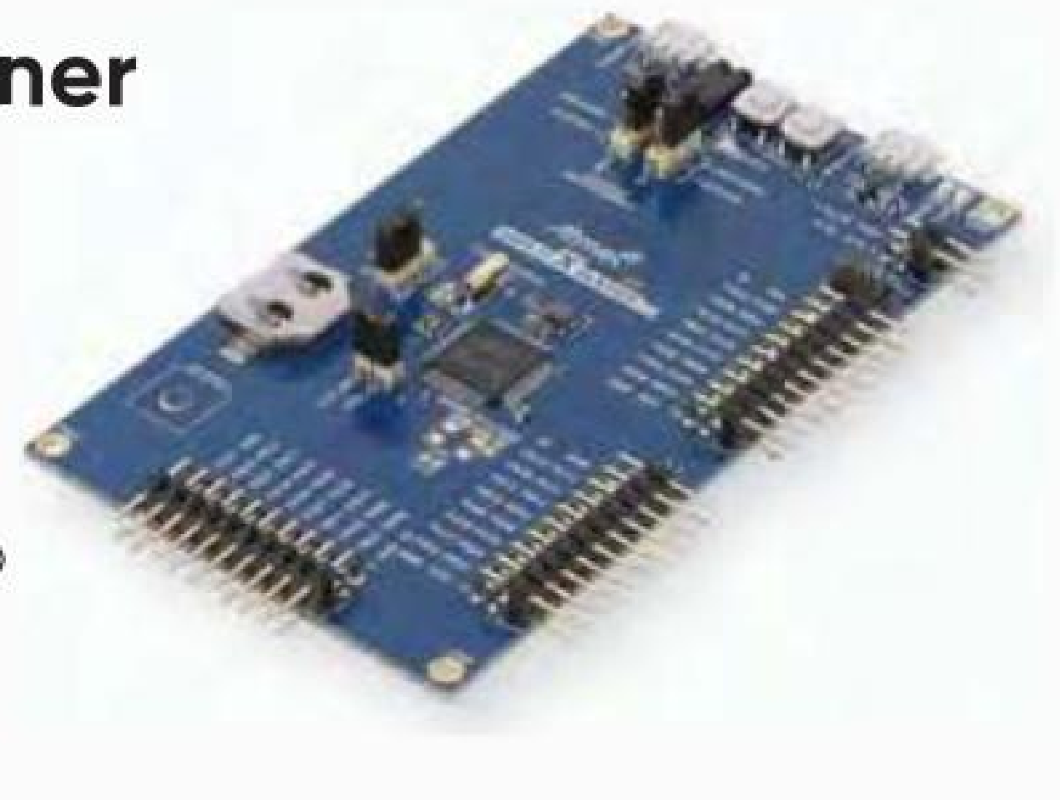
Closing date

The closing date for this offer is 30 November 2019.

July 2019 winner

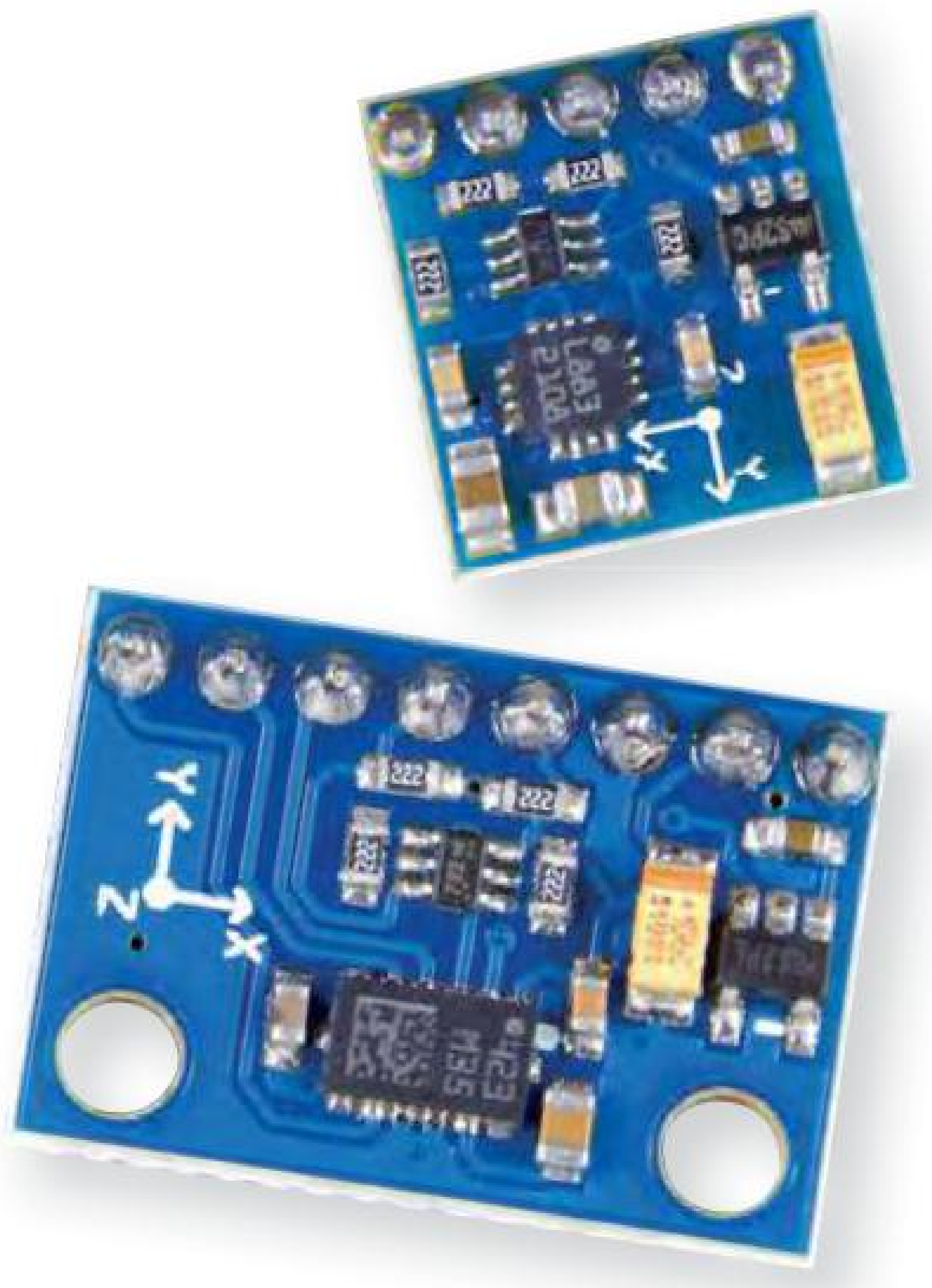
David Widdup

He won a Microchip SAM L21 Xplained Pro Evaluation Kit worth £36.75



Using Cheap Asian Electronic Modules Part 20: by Jim Rowe

Two tiny Electronic Compass modules



The Elecrow GY-271 and the GY-511 are two low-cost electronic compass modules. Both readily available modules incorporate a 3-axis magnetometer, with the GY-511 also including an accelerometer.

You can use them with an Arduino, Micromite or any other microcontroller which supports the I^{2}C communication protocol.

The GY-271 is the smaller of the two modules, measuring only 14.5 × 13.3 × 3.5mm (without the 5-pin header attached). It's based on the Honeywell HMC5883L 3-axis digital compass (magnetometer) IC, which is no longer being manufactured, but is nevertheless still available in significant quantities.

The GY-511 is nearly double the size, measuring 21 × 14.5 × 3.5mm (without the 8-pin header attached). This module is based on the STMicroelectronics LSM303DLHC 3D accelerometer/3D magnetometer IC, which is somewhat more complex than the HMC5883L.

A functionally identical clone of the GY-271 is available from Altronics (Cat Z6391) and Jaycar (Cat XC4496). This has a six-pin header rather than five, with the extra pin being a 3.3V output from the on-board regulator which you can use to power external circuitry. Since that connection is purely for convenience, the description of the GY-271 here applies to those modules too.

The GY-511 is also available from Altronics, Cat Z6391A. Interestingly, while the GY-511 is often a bit more expensive, Altronics charge exactly the same for it as they do the GY-271 clone. Given the extra functionality, that seems like the one to get.

The HMC5883L

The HMC5883L IC used in the GY-271 module comes in a tiny 3 × 3 × 0.9mm 16-pin LCC (leadless chip carrier) surface-mount package. A simplified version of its internal block diagram is shown in Fig.1.

There are actually two chips inside the HMC5883L: the sensing block on the far left (pink shading) which does the actual magnetic field sensing, and the measurement and control circuitry which forms the rest of the device. Presumably, this is necessary because they use different manufacturing processes.

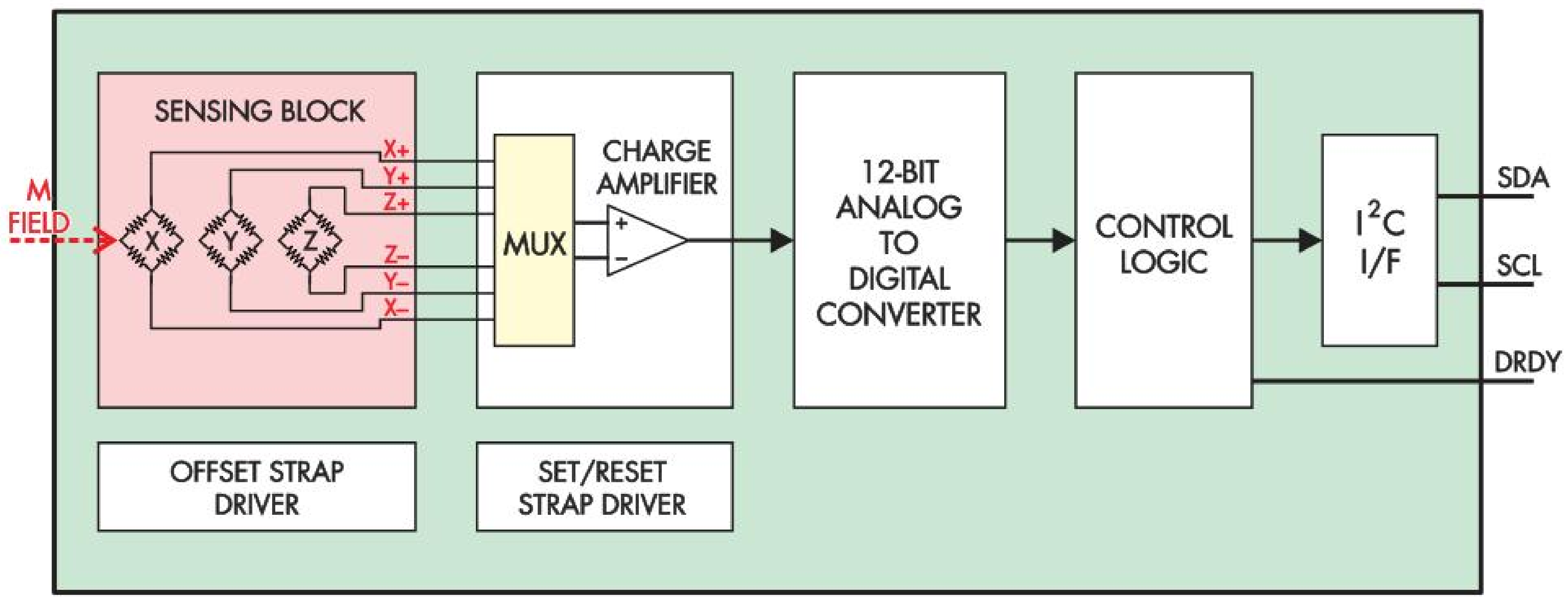
The sensing block chip has three magneto-resistive sensor bridges, oriented at right angles to each other. They are labelled X, Y and Z. This allows it to sense both the direction and magnitude of very low-intensity magnetic fields, like the one generated by the Earth.

The sensor bridge outputs are connected to the inputs of an analogue multiplexer (MUX) on the measurement chip, which allows the control circuitry to select them in turn.

The selected bridge output is then passed via a charge amplifier to the input of a 12-bit ADC (analogue-to-digital converter), which delivers its corresponding digital value to the control logic section.

When all three measurements have been made in this way, the control logic makes them available to an external MCU via the standard I^{2}C interface at far right.

The other two circuit blocks, labelled 'Offset Strap Driver' and 'Set/Reset Strap Driver' are used by the chip's control logic to perform degaussing, testing and offset compensation for the magneto-resistive sensor bridges. As a result, the device can offer magnetic field resolution down to 200nT (nanoTesla) or 2mG (milliGauss).



HMC5883L eCOMPASS DEVICE INTERNAL BLOCK DIAGRAM

Fig.1: block diagram for the Honeywell HMC5883L eCompass IC, showing the magnetic sensing bridges at upper left, which are connected to the charge amplifier by a multiplexer.

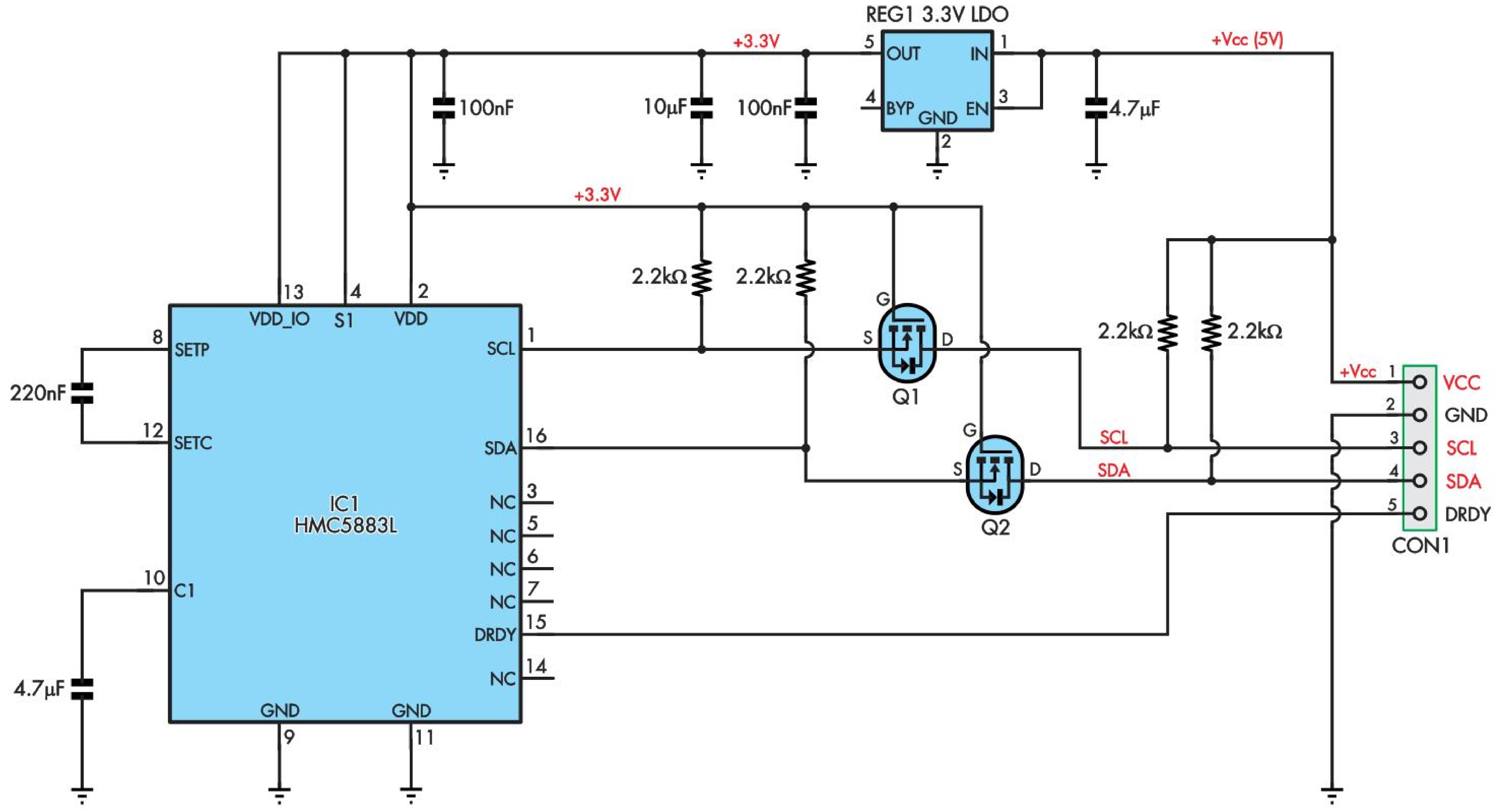


Fig.2: circuit diagram of the GY-271, which is based around the HMC5883L IC. It has few other components; primarily, voltage regulator REG1, level-shifting MOSFETs Q1 and Q2, some bypass/filtering capacitors and pull-up resistors.

This makes it very suitable for measurements of the Earth's magnetic field, which tends to vary between about 22µT and 64µT (microTesla) over the planet's surface. And it can make these measurements at a rate of up to 160Hz.

The supply current of the HMC5883L is very low, varying from around 2µA in idle mode up to about 100µA when it's making measurements. This makes it suitable for portable and hand-held applications like smartphones and tablets.

The circuit diagram of the complete GY-271 eCompass module is shown in Fig.2, with the HMC5883L forming the heart of this module. The only other active devices are REG1, a 3.3V LDO (low-dropout) regulator and N-channel MOSFETs Q1 and Q2, which perform level translation on the SCL and SDA lines of the module's I^{2}C interface.

This means that the HMC5883L can operate from a 3.3V supply rail but still exchange data with an external micro running from a 5V supply. In fact, the I^{2}C pull-up resistors (2.2kΩ) for CON1 connect to the incoming 5V supply.

The 220nF capacitor between pins 8 and 12 of IC1 determines its Set/Reset timing, while the 4.7µF capacitor from pin 10 to ground acts as a reservoir for the charge amplifier ahead of the ADC. Pin 15 provides a data ready signal at the end of each measurement cycle. This is brought out to pin 5 of CON1, for optional use by the MCU to which it's connected.

We'll describe how to use this module a bit later. First, let's take a look at the IC used in its larger sibling, the LCM303DLHC.

The LSM303DLHC IC

Fig.3 shows a simplified block diagram of the LSM303DLHC eCompass IC, and as you can see, it is a little more complex than the HMC5883L (Fig.1). Most of the additional complexity is because this device incorporates a 3-axis linear accelerometer as well as the 3-axis magnetometer.

The magnetometer's sensing system is similar to that in the HMC5883L, with three magneto-resistive sensor bridges oriented at right angles to each other.

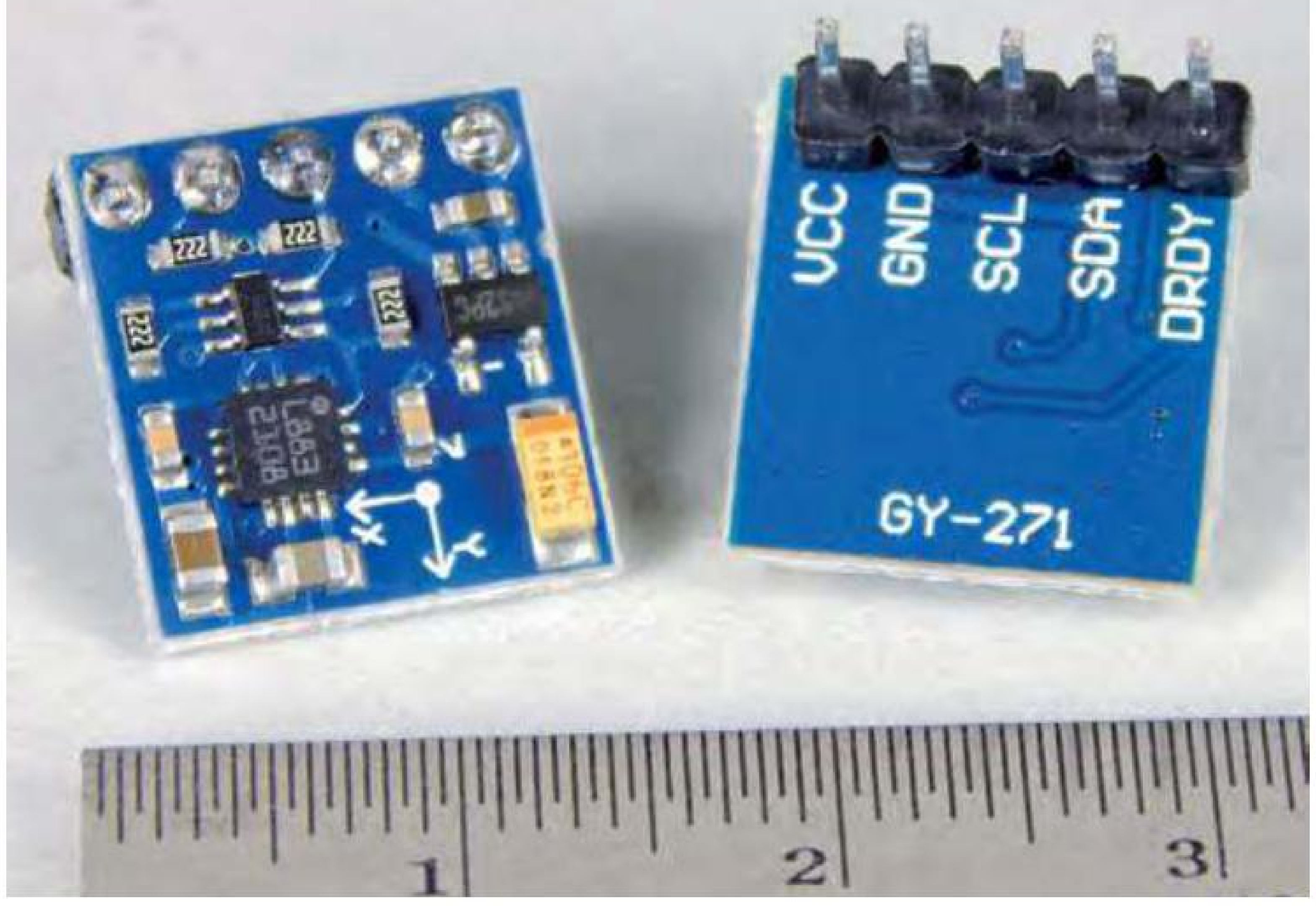
The linear accelerometer sensors are made from very thin micromachined strips, again oriented in mutually orthogonal directions, which cause capacitance changes when they deflect in response to any acceleration forces. They can also be used to sense gravitational fields, which allows the Earth's gravitational field to be used for calibrating the magnetometer.

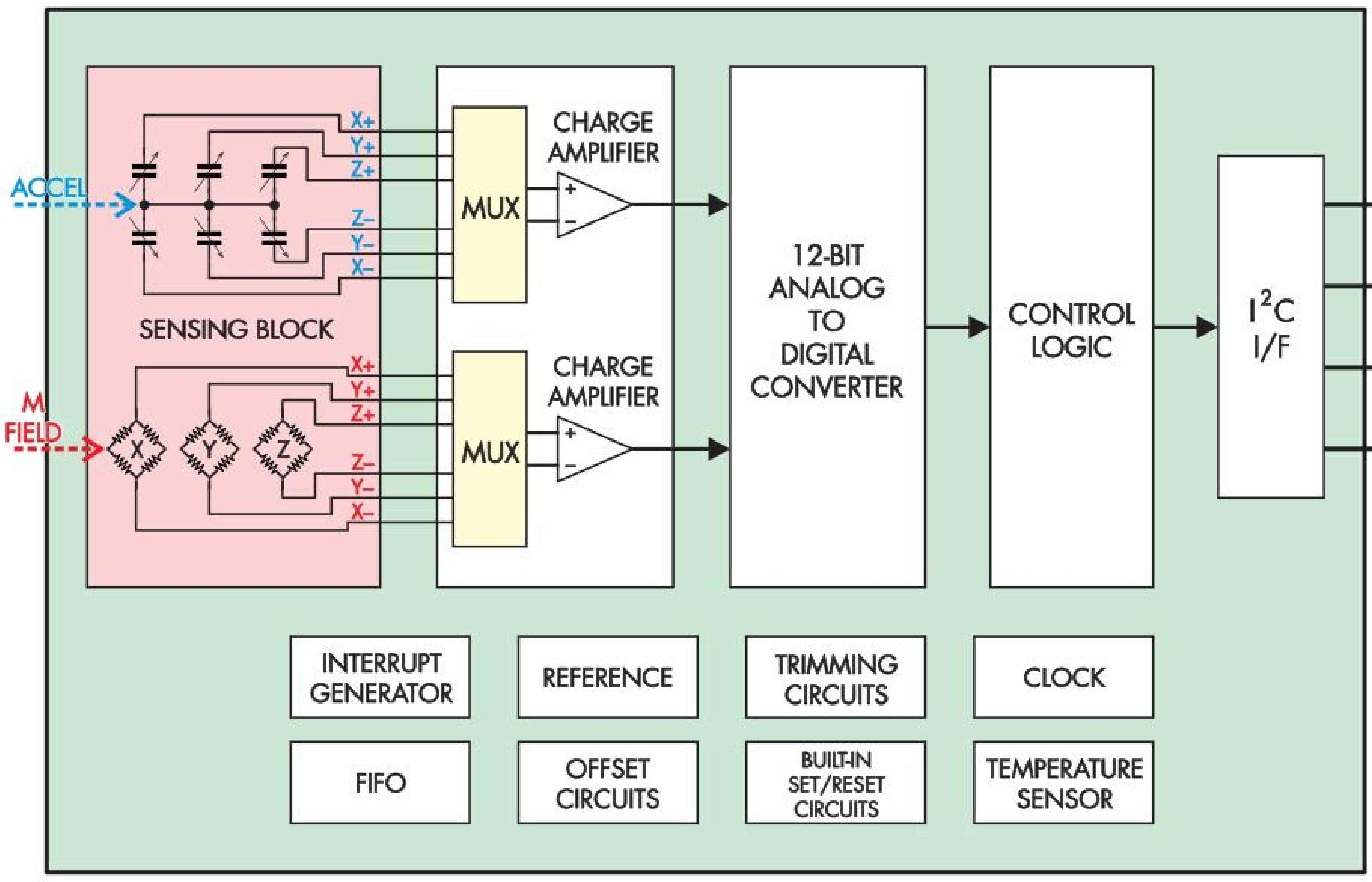
Both sensor arrays are shown in the pink shaded area of Fig.3, and they each have their own multiplexer and charge amplifier feeding the in-built ADC. The only other real differences from the HMC5883L are the additional blocks shown at the bottom of Fig.3.

Either of the two sensing arrays can be enabled or disabled by the control logic, in response to commands sent from the host MCU via the I^{2}C interface.

Since the accelerometer array is not really needed when you want to use the device as a simple eCompass, it can therefore be disabled. So when used as an eCompass, the LSM303DLHC is quite similar to the HMC5883L.

The LSM303DLHC draws about 110µA in normal measurement mode and around 1µA in idle/sleep mode. It has seven magnetic measurement ranges, varying from ±1.3 gauss to ±8.1 gauss (1G = 100µT), a maximum magnetic resolution of 2mG (0.2µT or 200nT) and the ability to make measurements at eight selectable rates, from 0.75Hz to 220Hz.




ACCEL
SENSING BLOCK
M FIELD
X+
Y+
Z+
Z–
Y–
X–
X
Y
Z
MUX
MUX
CHARGE AMPLIFIER
CHARGE AMPLIFIER
12-BIT ANALOG TO DIGITAL CONVERTER
CONTROL LOGIC
I²C I/F
SDA
SCL
INT1
INT2
INTERRUPT GENERATOR
REFERENCE
TRIMMING CIRCUITS
CLOCK
FIFO
OFFSET CIRCUITS
BUILT-IN SET/RESET CIRCUITS
TEMPERATURE SENSOR

LSM303DLHC 3D ACCELEROMETER/MAGNETOMETER INTERNAL BLOCK DIAGRAM

Fig.3: the STMicro LSM303DLHC IC is similar to the HMC5883L shown in Fig.1 but also incorporates a three-axis MEMS accelerometer along with an additional multiplexer and amplifier. This allows the compass' orientation to be determined, for more accurate results.

So once again, the LSM303DLHC IC forms the heart of the GY-511 eCompass module, as shown in Fig.4. If you compare this with Fig.2, you'll see that they're almost identical.

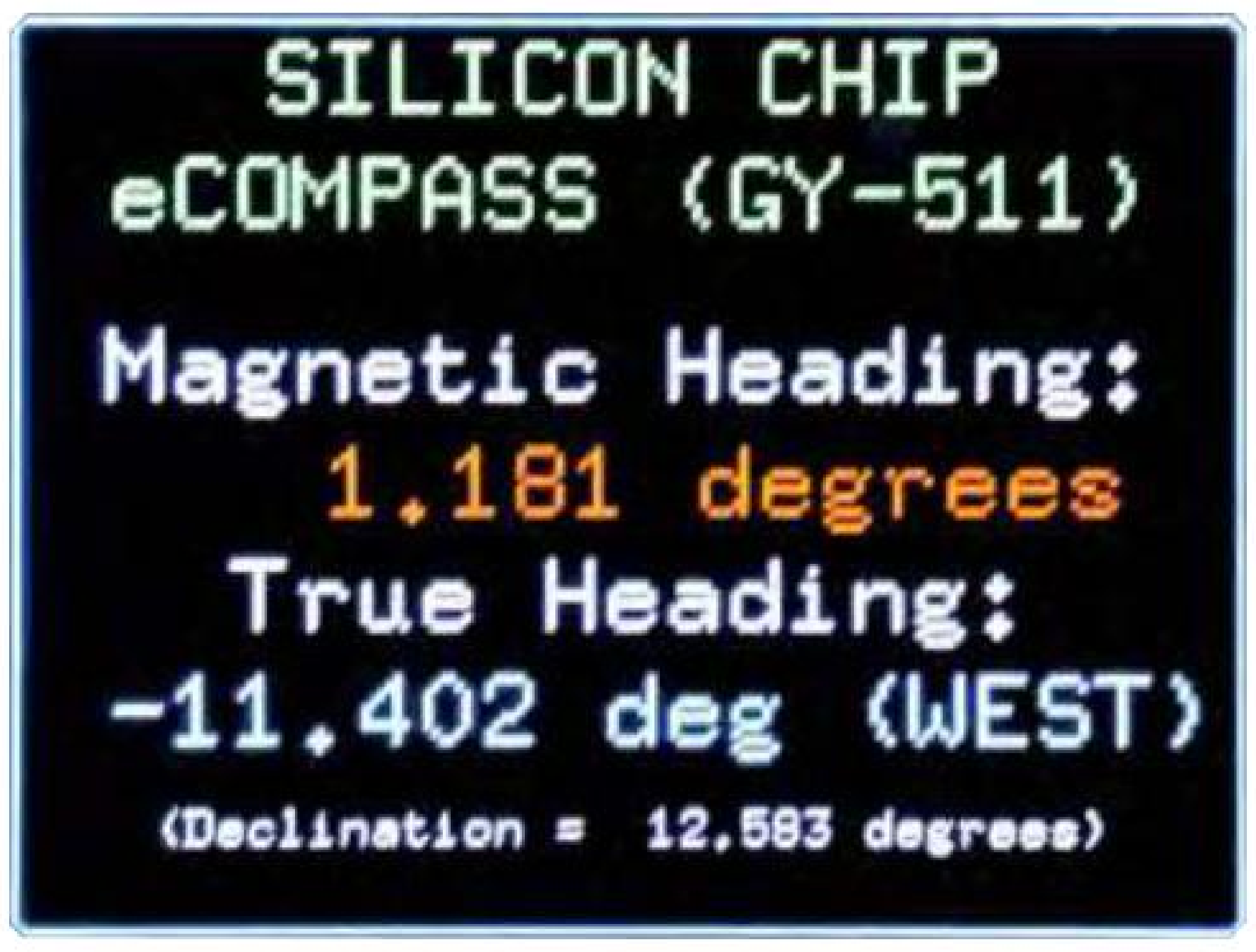
The only differences are the chip for IC1 and an 8-pin header for CON1 instead of a 5-pin header. When using the GY-511 module as an eCompass, the additional pins can be ignored.

Connecting to a micro

As both modules use an I^{2}C serial interface to exchange data with an MCU, connecting them to an Arduino or a Micromite is straightforward. Fig.5 shows how a GY-271 is connected to an Arduino, while Fig.6 shows how it's connected to a Micromite. Similarly, Fig.7 shows how a GY-511 module is connected to an Arduino, while Fig.8 shows how it's connected to a Micromite.

Things are not quite so straightforward when it comes to the software. You would expect that there are already Arduino libraries suitable for interfacing with these modules, and indeed they are available. But when I tried them out, I found most of them to be too complicated, poorly written and/or buggy.

The only library I found that was both easy to use and worked well was one called **HMC5883L_Simple**, written by James Sleeman in New Zealand. This library can be downloaded from Mr Sleeman's website, at: **http://sparks.gogo.co.nz/HMC5883L_Simple.zip**



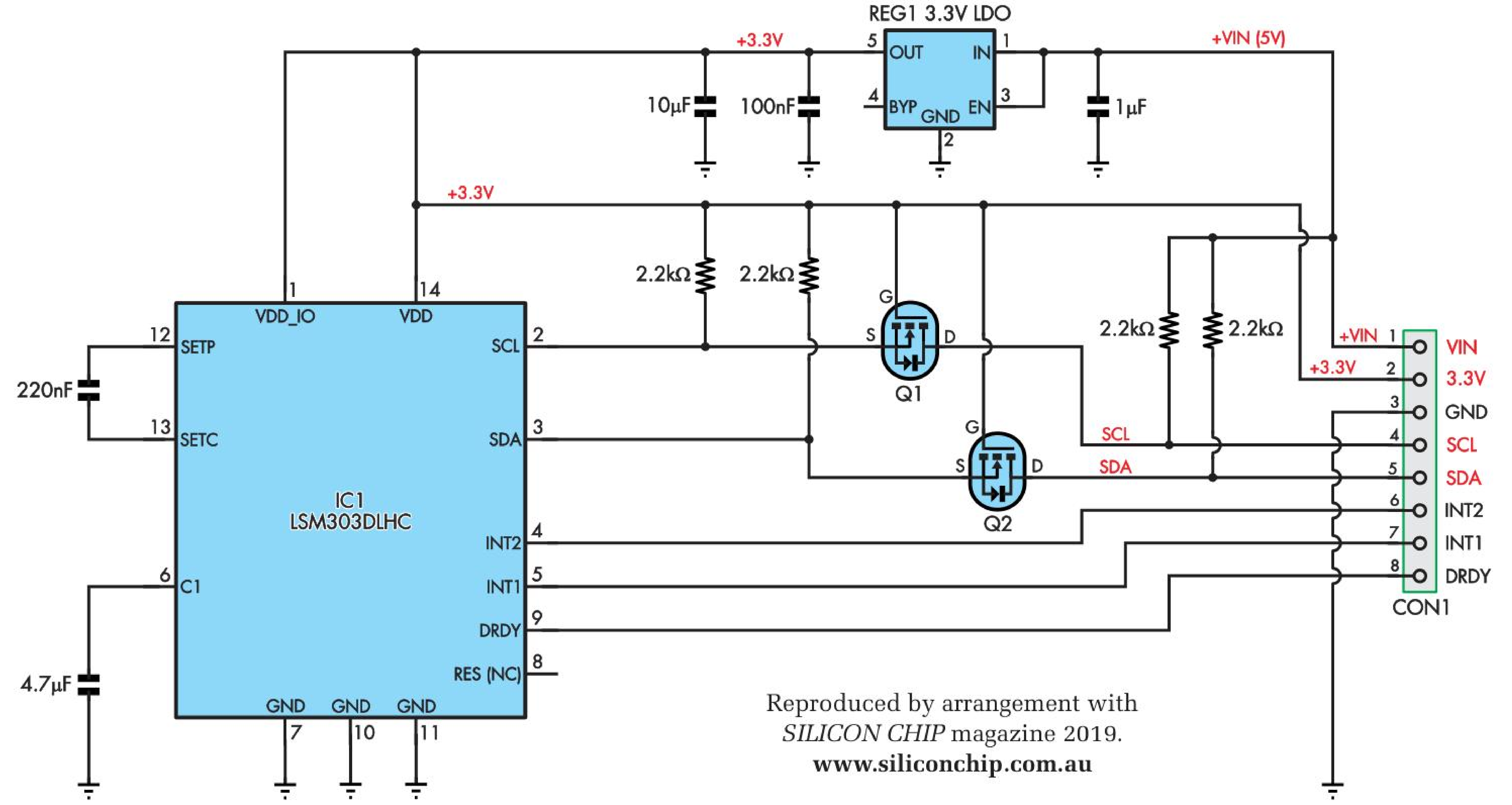
SILICON CHIP
eCOMPASS (GY-511)
Magnetic Heading:
1.181 degrees
True Heading:
-11.402 deg (WEST)
(Declination = 12.583 degrees)

Our example MMBasic program shows both the magnetic heading (relative to the north magnetic pole) and the true heading (relative to the north celestial pole).

The archive file includes a simple example sketch (**Compass.ino**), which I can recommend. A sample grab of the Arduino IDE's Serial Monitor output when running this sketch is shown opposite, with the GY-271 module's Y axis pointing to magnetic north. The heading figures are all within the range of 0.43-2.14°N.

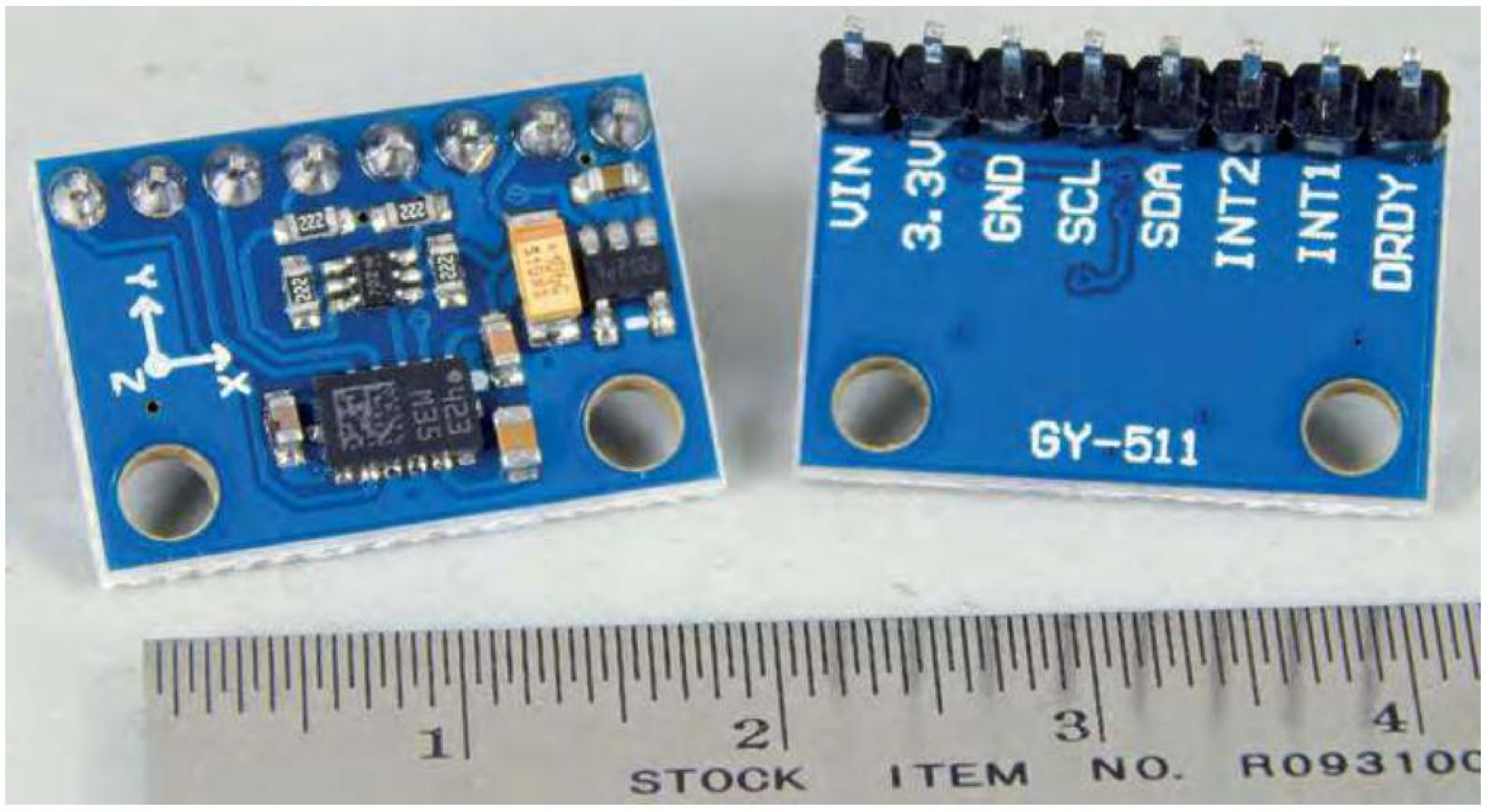
Since the two modules are similar, we adapted this library to work with the GY-511 module without any modifications, although the magnitude of the results may be wrong (this isn't terribly important when using it as a compass).

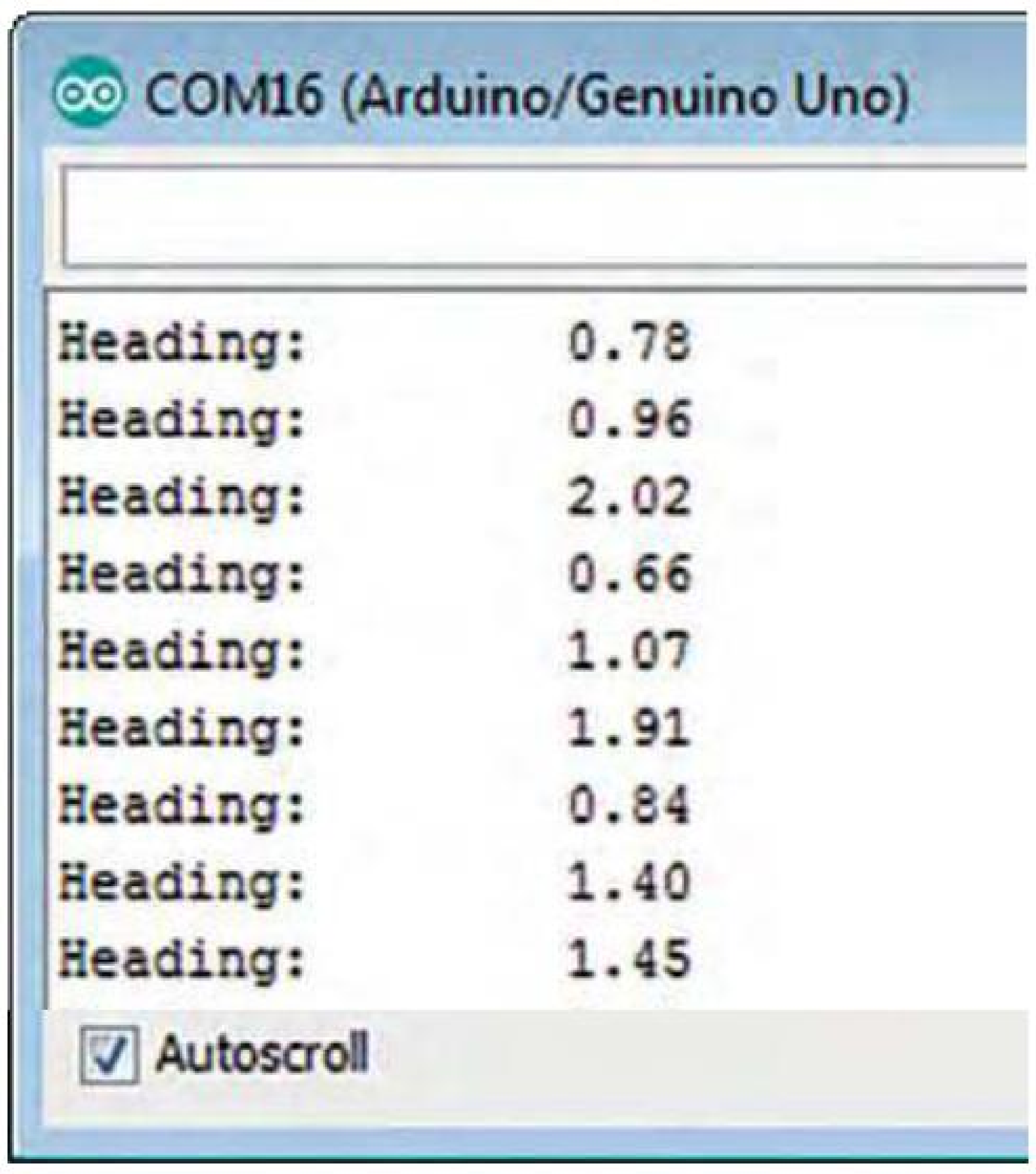
When it comes to using either module with a Micromite, I couldn't find any existing programs or libraries.


REG1 3.3V LDO
+3.3V
+VIN (5V)
OUT
IN
BYP
GND
EN
10µF
100nF
1µF
+3.3V
2.2kΩ
2.2kΩ
2.2kΩ
2.2kΩ
G
S
D
Q1
Q2
VDD_IO
VDD
SETP
SETC
SCL
SDA
IC1
LSM303DLHC
INT2
INT1
C1
DRDY
RES (NC)
GND
GND
GND
220nF
4.7µF
+VIN
+3.3V
SCL
SDA
VIN
3.3V
GND
SCL
SDA
INT2
INT1
DRDY
CON1

Reproduced by arrangement with *SILICON CHIP* magazine 2019.
www.siliconchip.com.au

Fig.4: the circuit for the GY-511 eCompass module, which is virtually identical to the GY-271 shown in Fig.2, except that a different IC is used and it has two extra interrupt signal connections which are wired to header CON1. This has more pins (eight, compared to five), along with a 3.3V output from REG1.





```
Heading:        0.78
Heading:        0.96
Heading:        2.02
Heading:        0.66
Heading:        1.07
Heading:        1.91
Heading:        0.84
Heading:        1.40
Heading:        1.45
```

The GY-511 module shown enlarged above and to its right is the example serial output from the Compass.ino sketch (James Sleeman's Arduino library) using the GY-271.

So I had to analyse the functions embedded in Arduino libraries (especially Mr Sleeman's), and then write MMBasic programs to duplicate the same functions on the Micromite. The programs I wrote are called **GY271 eCompass.bas** and **GY511 eCompass.bas** and both are available from the November 2019 page of the *PE* website. Note that all of these programs (Arduino and Micromite) treat the Y-axis of the module as the 'needle' of the eCompass.

These programs do the bare minimum to allow the modules to be used as electronic compasses. They initialise the main IC, then make measurements twice a second, process the X and Y data readings to arrive at the magnetic heading, then convert this to a true heading by subtracting the local declination figure.

How a compass works

A compass is a portable device used to work out your heading. This is the direction in which you are travelling with respect to the Earth's axis of rotation, or the hypothetical meridian lines on the surface of the Earth between the true south and north poles. It does this by sensing the weak magnetic field which surrounds the Earth, due to the magnetisation of the Earth's metal core.

A traditional compass senses the Earth's field by means of a small magnetised iron needle which is able to rotate freely in the horizontal plane about its centre because it's either floating on a small pool of liquid or mounted at its centre on a very low friction needle bearing.

As a result, the needle can orient itself to align with the horizontal component of the Earth's magnetic field, so the needle always tends to point towards north. A dial around the circumference of the compass then allows the user to work out the direction of any desired heading.

That's the basic idea, anyway. But in practice, things are a little more complicated. That's because while the compass needle aligns itself with the Earth's magnetic field passing from south to north, that field passes between the Earth's magnetic poles and these are different from the Earth's true geographic poles (which correspond to its axis of rotation).

Not only that, but the magnetic field is not uniform with smooth meridian lines passing between the south and north magnetic poles. In fact, the field lines weave around quite a bit, with an orientation varying significantly according to latitude and longitude and also according to time, as the field pattern changes from year to year.

So wherever you happen to be, although the needle of a compass nominally points towards north, that doesn't mean that it shows the direction of true north. To work out the direction of true north, you need to know the angle between the horizontal component of the Earth's magnetic field at that location and a meridian line from the true south pole to the true north pole at the same location.

This angle is called the 'magnetic declination' and you can find the declination at any particular point on the Earth's surface by referring to either maps or websites like www.magnetic-declination.com

The declination varies quite significantly over Australia and New Zealand. For example, in Sydney, it's around 12.6°E while in Perth it's around 1.8°W. The current declinations for a number of locations in Oceania are shown in Table 1.

There's another aspect of the Earth's magnetic field that can affect compass operation. That's the fact that the magnetic field at any particular location is not aligned parallel to the Earth's surface (ie, in the horizontal plane) but in many places is at a significant angle.

This is called the 'magnetic inclination', and broadly speaking (when facing north) it points down into the ground in the northern hemisphere and upwards away from the ground in the southern hemisphere. This doesn't have a major effect on compass operation, but it sometimes does need to be taken into account, especially with traditional compasses.

Table 1 also shows the inclination of the Earth's magnetic field for each location. All the inclinations listed are oriented upwards (because all locations are in the southern hemisphere) but they vary with latitude. The locations that are furthest south have a noticeably higher inclination than those nearer the equator.

TABLE 1: Magnetic Declination & Inclination figures in our area

LOCATION	DECLINATION	INCLINATION
Adelaide SA	+8° 1'	66° 59'
Alice Springs NT	+4° 34'	55° 31'
Brisbane QLD	+11° 3'	57° 35'
Broome WA	+1° 46'	48° 34'
Cairns QLD	+6° 47'	45° 22'
Christmas Island	+0° 22'	38° 05'
Darwin NT	+2° 38'	39° 44'
Hobart TAS	+15° 10'	72° 38'
Melbourne VIC	+11° 41'	68° 49'
Onslow WA	+0° 32'	54° 22'
Perth WA	–1° 50'	66° 08'
Rockhampton QLD	+9° 15'	53° 19'
Sydney NSW	+12° 35'	64° 18'
Auckland NZ	+19° 58'	62° 51'
Christchurch NZ	+24° 10'	68° 38'
Dunedin NZ	+25° 47'	70° 46'
Invercargil NZ	+25° 44'	71° 32'
Wellington NZ	+22° 43'	66° 32'
Dili TIMOR-LESTE	+1° 28'	32° 56'
Jakata INDONESIA	+0° 38'	29° 49'
Kuala Lumpur MALAYSIA	+0° 1'	9° 49'
Port Moresby PNG	+5° 53'	33° 22'

NOTE: Positive declination = –> EAST, negative = –> WEST
Inclination is UPWARD towards north in Southern Hemisphere

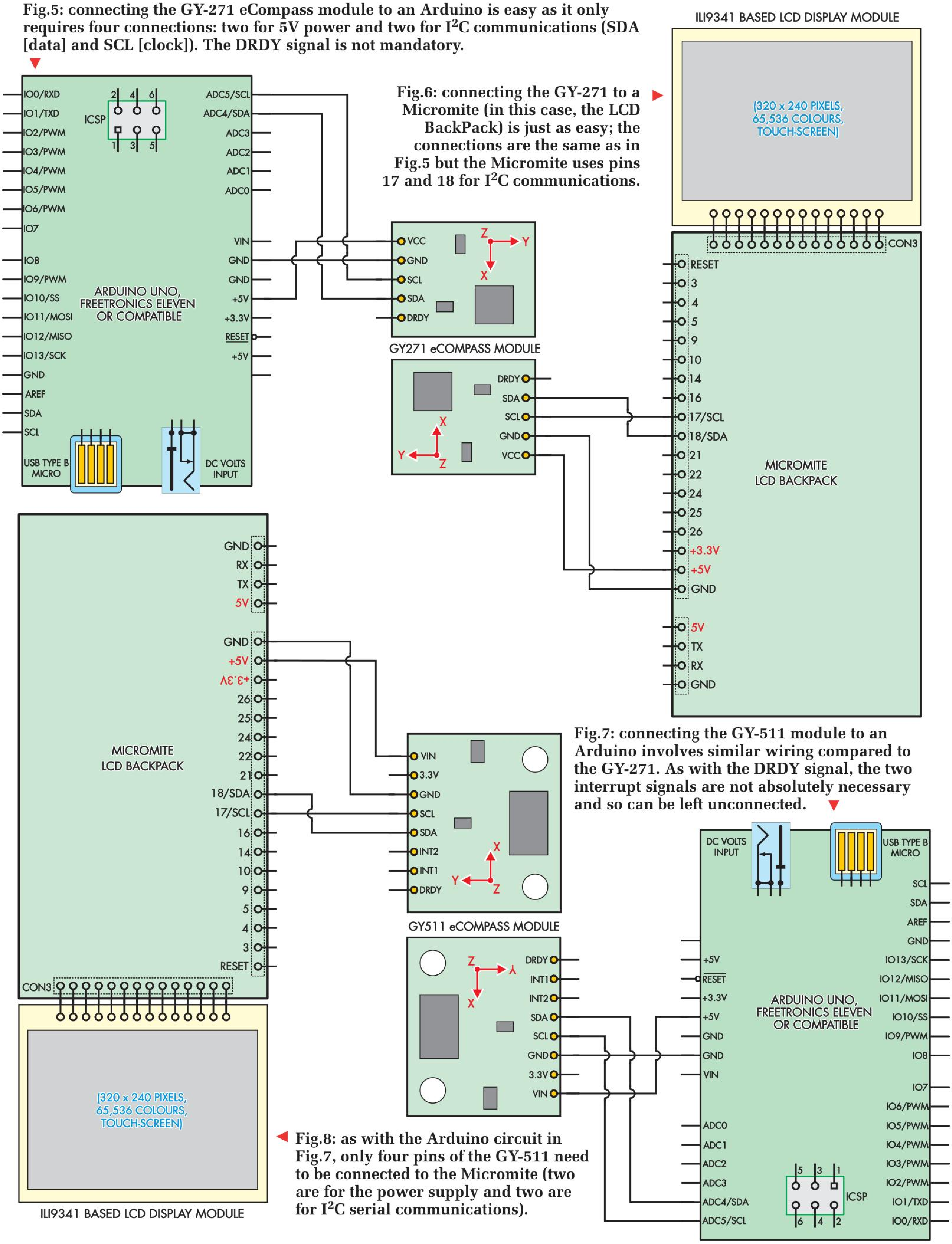

ILI9341 BASED LCD DISPLAY MODULE
(320 x 240 PIXELS, 65,536 COLOURS, TOUCH-SCREEN)
CON3
IO0/RXD
IO1/TXD
IO2/PWM
IO3/PWM
IO4/PWM
IO5/PWM
IO6/PWM
IO7
IO8
IO9/PWM
IO10/SS
IO11/MOSI
IO12/MISO
IO13/SCK
GND
AREF
SDA
SCL
ICSP
ADC5/SCL
ADC4/SDA
ADC3
ADC2
ADC1
ADC0
VIN
GND
+5V
+3.3V
RESET
ARDUINO UNO, FREETRONICS ELEVEN OR COMPATIBLE
USB TYPE B MICRO
DC VOLTS INPUT
VCC
SCL
SDA
DRDY
X
Y
Z
GY271 eCOMPASS MODULE
MICROMITE LCD BACKPACK
3
4
5
9
10
14
16
17/SCL
18/SDA
21
22
24
25
26
+3.3V
+5V
5V
TX
RX
VIN
3.3V
INT2
INT1
GY511 eCOMPASS MODULE

Fig.5: connecting the GY-271 eCompass module to an Arduino is easy as it only requires four connections: two for 5V power and two for I^{2}C communications (SDA [data] and SCL [clock]). The DRDY signal is not mandatory.

Fig.6: connecting the GY-271 to a Micromite (in this case, the LCD BackPack) is just as easy; the connections are the same as in Fig.5 but the Micromite uses pins 17 and 18 for I^{2}C communications.

Fig.7: connecting the GY-511 module to an Arduino involves similar wiring compared to the GY-271. As with the DRDY signal, the two interrupt signals are not absolutely necessary and so can be left unconnected.

Fig.8: as with the Arduino circuit in Fig.7, only four pins of the GY-511 need to be connected to the Micromite (two are for the power supply and two are for I^{2}C serial communications).

Both heading figures are then displayed on the Micromite's LCD screen, as you can see from the screenshot. Note that the current declination is also shown at the bottom of the screen, as a reminder. The declination adjustment is necessary because the Earth's magnetic north pole is not at the actual north pole; in fact, they are getting further apart each year, so you may need to update this value occasionally to maintain accuracy. See the *How a* *compass works* panel for more detail on the differences between magnetic north and true north.

Both programs are written to include the magnetic declination of Sydney (12.583°E, as shown in the previous screenshot). If you're at a different location, you need to modify the source code to include the correct declination value for your location, near the start of the program:

```
DIM AS FLOAT Declin! = 12.583
```

Like Mr Sleeman's Arduino library and example sketch, my Micromite programs make no allowance for the local inclination (tilt) of the Earth's magnetic field. In this respect, they are the same as a traditional compass – both programs assume that the module's PCB (and thus its magnetometer chip) is being held in the horizontal plane or close to it.

It is possible to take the magnetic inclination into account when working out the absolutely true heading of an eCompass, but you need to combine the data from the magnetometer with data from an accelerometer or gravity field detector like the one in the LSM303DLHC chip. So you could not do this with the GY-271 module unless you also had a separate accelerometer.

Handy links

HMC5883L datasheet:
www.farnell.com/datasheets/1683374.pdf

LSM303DLHC datasheet:
www.st.com/resource/en/datasheet/DM00027543.pdf

Magnetic declination:
https://en.wikipedia.org/wiki/Magnetic_declination

Geomagnetic declination:
www.ga.gov.au/oracle/geomag/agrfform.jsp

Tag-Connect

JTAG Connector Plugs Directly into PCB!!

No Header! No Brainer!

Our patented range of Plug-of-Nails™ spring-pin cables plug directly into a tiny footprint of pads and locating holes in your PCB, eliminating the need for a mating header. Save Cost & Space on Every PCB!!

Solutions for: PIC . dsPIC . ARM . MSP430 . Atmel . Generic JTAG . Altera Xilinx . BDM . C2000 . SPY-BI-WIRE . SPI / IIC . Altium Mini-HDMI . & More

www.PlugOfNails.com

Tag-Connector footprints as small as 0.02 sq. inch (0.13 sq cm)

Teach-In 8 CD-ROM
Exploring the Arduino

This CD-ROM version of the exciting and popular *Teach-In 8* series has been designed for electronics enthusiasts who want to get to grips with the inexpensive, immensely popular Arduino microcontroller, as well as coding enthusiasts who want to explore hardware and interfacing. *Teach-In 8* provides a one-stop source of ideas and practical information.

The Arduino offers a remarkably effective platform for developing a huge variety of projects; from operating a set of Christmas tree lights to remotely controlling a robotic vehicle wirelessly or via the Internet. *Teach-In 8* is based around a series of practical projects with plenty of information for customisation. The projects can be combined together in many different ways in order to build more complex systems that can be used to solve a wide variety of home automation and environmental monitoring problems. The series includes topics such as RF technology, wireless networking and remote web access.

PLUS: PICs and the PICkit 3 – A beginners guide

The CD-ROM also includes a bonus – an extra 12-part series based around the popular PIC microcontroller, explaining how to build PIC-based systems.

SOFTWARE

The CD-ROM contains the software for both the *Teach-In 8* and PICkit 3 series.

PRICE
£8.99

Includes P&P to UK if ordered direct from us

ORDER YOUR COPY TODAY!
JUST CALL 01202 880299 OR VISIT www.epemag.com

Tinnitus & Insomnia Killer

by
John Clarke

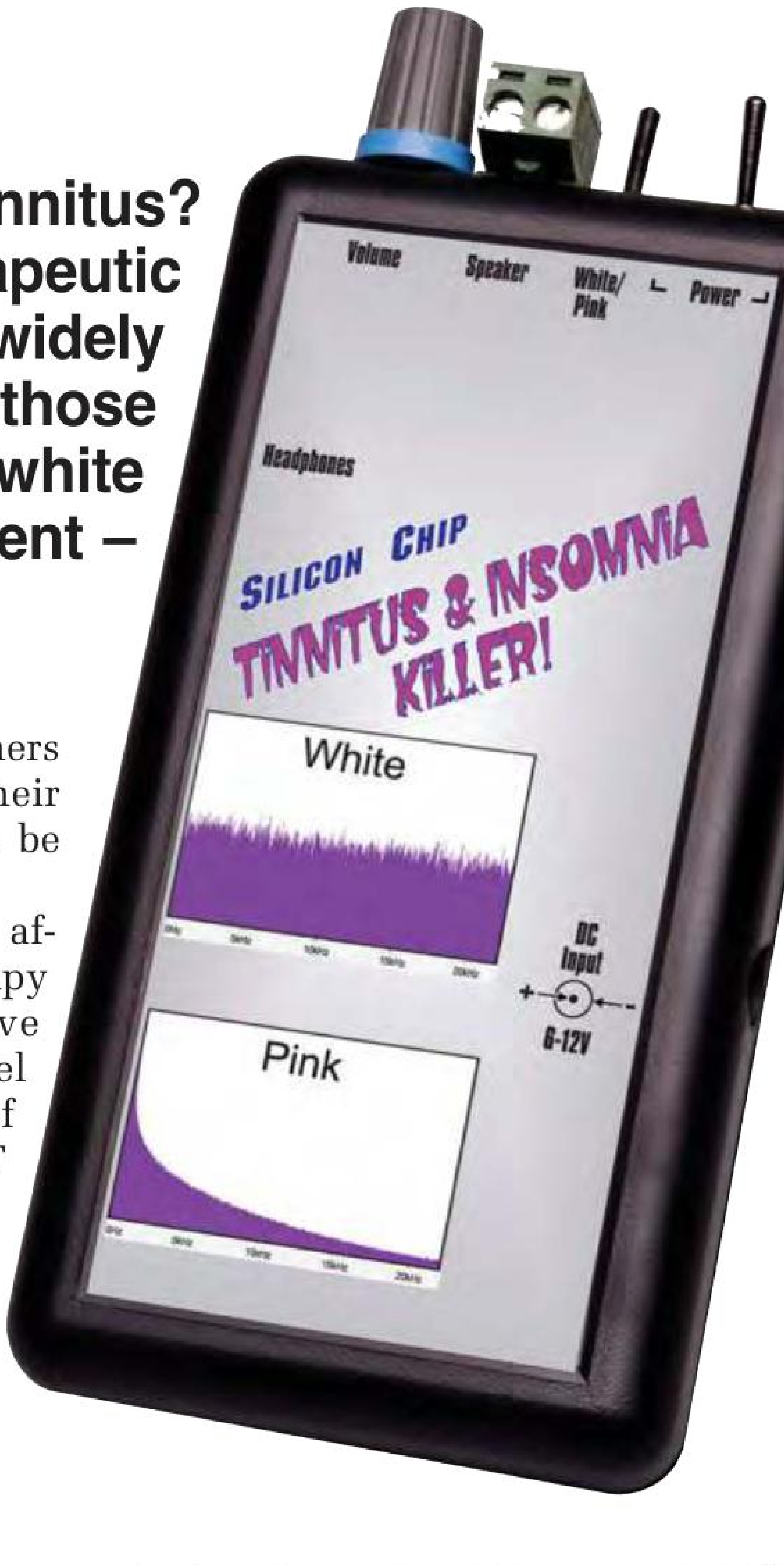
Do you – or someone you know – suffer from Tinnitus? How about Insomnia? We can't make any therapeutic promises, but pink and/or white noise is widely recognised as easing or even eliminating those problems! This device produces either pink or white noise so you can experiment to your ears' content – and maybe get some relief!

If you have never suffered from Tinnitus, consider yourself fortunate! Tinnitus is the perception of sound when no external sound is present.

Commonly referred to as 'ringing in the ears', Tinnitus may sound like humming, clicking, buzzing, ringing, hissing, roaring, whistling or even the sound of crickets.

It is especially maddening for those who suffer from it constantly. Tinnitus may be intermittent or constant, and it may vary in loudness depending on stress levels, blood pressure, tiredness, medications and the surrounding environment. Some people who experience Tinnitus are not really bothered by it. But others find that it seriously disturbs their sleep. In the worst case, it can be debilitating.

For people who are severely affected, Tinnitus Retraining Therapy (TRT) can provide an effective treatment. Developed by Dr Jawel Jastreboff, TRT involves the use of low-level broadband noise. TRT



The *Tinnitus & Insomnia Killer*, housed in a Jaycar handheld instrument case. (The Altronics case version is similar).

does not cure Tinnitus, but it does make it manageable for people who are severely affected.

If you want to know more, there are many websites devoted to Tinnitus. Just call up 'Tinnitus' in your favourite search engine and you will find lots of information.

Even if you don't suffer from this affliction, there are times when a low-level noise source can be really helpful in masking extraneous noise – such as when you can't get to sleep and lie there tossing and turning, getting even more worked up and even less likely to find that elusive sleep!

Insomnia

Perhaps a nearby neighbour is having a boisterous party and counting sheep or hiding your head under your pillow simply doesn't work! Switch on the *Tinnitus & Insomnia Killer* and you can effectively blank out the noise that's keeping you awake. Or maybe you are trying to study and someone else in the family has swapped their ballet shoes for hob-nail boots. Again, switch on this unit and mask it. Our only warning is that if you're trying to study, you may instead fall asleep. Oh well, can't win 'em all!

How does it do this?

The *Tinnitus & Insomnia Killer* masks external sounds by increasing the ambient noise level so that the unwanted noise is much less obtrusive.

The 'noise' from this unit is something you can live with – in fact, it is often quite soothing. It has been likened to what you hear from light rain on a tin roof, a soft waterfall or a stream cascading down rocks.

Babies experience constant noise like this in the womb, which is why 'shushing' them often calms them down and helps them get to sleep.

We're all accustomed to this sort of noise when we are very young. So it can be especially effective at helping babies to sleep, even when they are in a quiet environment. To them, a quiet environment is quite an alien concept!

Finally, another use for white/pink noise: relaxation. There might be no doof-doof noise from the neighbour's party – in fact, it might be too quiet for you to relax. Just add a little noise (of the right type) and you'll probably find you can relax much more easily.

Light and compact

The *Tinnitus & Insomnia Killer* can drive headphones or a loudspeaker. And it can be powered from a 9V battery or a DC plugpack (from about 6-12V). It's built into a small plastic case and it includes a volume control to set the level that suits you.

White noise, pink noise: what's the difference?

White noise has equal energy at all frequencies across its entire bandwidth. So, for example, the 1kHz band from 1-2kHz will have the same total energy as the 1kHz band from 10-11kHz. In practice, this means that white noise has a 3dB rise in amplitude for each higher octave. It sounds similar to steam escaping or when an FM radio is tuned off-station. Pink noise, on the other hand, has an equal energy level for each octave. For example, the total energy in the 20-40Hz band (ie, 20Hz bandwidth) is the same as from 10-20kHz (10kHz bandwidth). Therefore, it has an identical amplitude for each octave.

In effect, this means that pink noise sounds more subdued and less harsh than white noise and has more apparent bass.

For Tinnitus suffers or those simply looking to mask out unwanted noise, whether you use white or pink noise comes down to your preference. Try them both and see which one you prefer and which is more effective in your situation.

Features and Specifications

- **Produces white or pink noise**
- **Onboard volume control**
- **Drives headphones, earbuds or a loudspeaker (up to 750mW into 8Ω)**
- **Powered from a 9V battery or 6-12V DC plugpack**
- **Power-on and low battery indicator**
- **Small, light and portable (handheld) unit**
- **Current draw with headphones: 4.6mA quiescent, 8-9mA# at medium volume, 20-25mA# at full volume**
- **Current draw with speaker: 8.6mA quiescent, 47-80mA# at full volume**
- **9V battery life: typically around 48 hours with headphones or 7 hours with speaker**

# lower figure is for white noise, slightly higher for pink noise

Other uses for pink noise

Besides helping those with Tinnitus or as a sleep aid, pink noise is often used in the laboratory – for measuring and testing loudspeaker systems, for example. It can be used when positioning and adjusting speakers to compensate for sound 'colouration' due to objects in the room and the shape of the room. It may also be used as a guide to get consistent sound throughout a room.

The pink noise is used as a signal source to drive the loudspeaker(s) via an amplifier. The resulting sound is monitored using a calibrated microphone; ie, one with a flat response, or a known response that can be compensated for.

The microphone drives a spectrum/frequency analyser to show how the sound changes as the microphone is moved around the room.

For room equalisation, an equaliser can be used to adjust the levels in each frequency band so that the overall frequency response is flat. *Our 10-band Graphic Equaliser* design from the June and July 2018 issues would be a good choice.

However, you don't need an expensive spectrum analyser for this job as there are many computer software-based options to display the audio spectrum from a microphone.

For example, there is a program called Wavespectra (**http://nice.kaze.com/av/wavespectra.html**). Another you might be familiar with is Audacity (**www.audacityteam.org/**). There are many others: Dr Google is your friend!

Circuit description

Refer now to the circuit diagram in Fig.1. IC1 is a PIC12F617 microcontroller which is programmed to produce white noise using a 31-bit pseudo-random noise sequence implemented in its software. It's called 'pseudo-random' because it's not truly random – the sequence repeats after about eight hours. So the repetition is not noticeable nor even statistically relevant in 99.9% of cases.

IC1's output signal passes to two different filters, one of which converts the white noise to pink (via IC2a) and one of which merely conditions the white noise further (via IC2b). Switch S2 determines which of these two signals is fed to amplifier IC3, allowing you to choose white or pink noise.

The internal 8MHz oscillator of the PIC12F617 is used, which gives a 2MHz instruction rate, so the 13 instructions in the software results in a sampling rate of 153.846kHz (2MHz÷13). The noise frequency distribution is therefore up to about half that, ie, 76.923kHz.

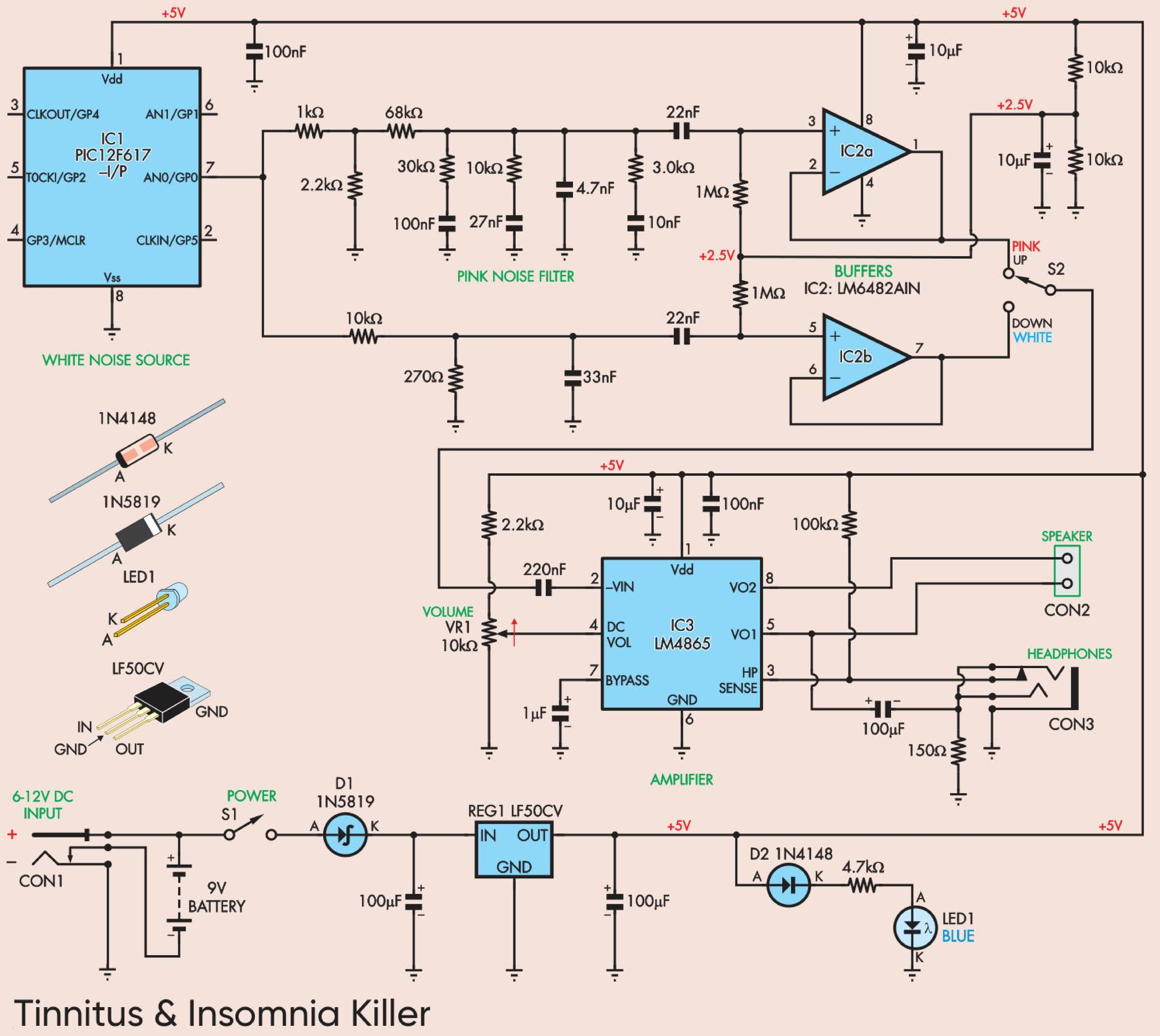

+5V
100nF
Vdd
IC1
PIC12F617
–I/P
CLKOUT/GP4
AN1/GP1
T0CKI/GP2
AN0/GP0
GP3/MCLR
CLKIN/GP5
Vss
WHITE NOISE SOURCE
1kΩ
68kΩ
2.2kΩ
30kΩ
10kΩ
100nF
27nF
4.7nF
3.0kΩ
10nF
PINK NOISE FILTER
22nF
1MΩ
+2.5V
10μF
10kΩ
IC2a
IC2b
BUFFERS
IC2: LM6482AIN
PINK
UP
S2
DOWN
WHITE
270Ω
33nF
1N4148
1N5819
LED1
LF50CV
IN
GND
OUT
2.2kΩ
220nF
VOLUME
VR1
10kΩ
–VIN
DC VOL
BYPASS
IC3
LM4865
VO2
VO1
HP SENSE
1μF
100kΩ
SPEAKER
CON2
HEADPHONES
CON3
100μF
150Ω
AMPLIFIER
6-12V DC INPUT
CON1
POWER
S1
9V BATTERY
D1
1N5819
REG1 LF50CV
D2 1N4148
4.7kΩ
LED1
BLUE
Tinnitus & Insomnia Killer

Fig.1: the circuit for the *Tinnitus & Insomnia Killer*. White noise is produced by IC1 at its pin 7 output. It is attenuated and buffered by op amp IC2b and then fed to IC3, when switch S2 is in the down position. The white noise is also converted to pink noise and buffered by IC2a and this is instead fed to IC3 if S2 is in the up position. IC3 amplifies the chosen signal and feeds it either to headphones at CON3, or to a speaker at CON2 if nothing is plugged into CON3.

Because the output is a square wave, it will have harmonic components at higher frequencies than 76.923kHz, but they will have a decreased amplitude and power level. The measured spectrum from IC1 is shown in Fig.4.

It extends over the entire audio spectrum (20Hz-20kHz) and well beyond at both the low-frequency and high-frequency ends.

Compare the spectrum of the pink noise produced by this unit, shown in Fig.2, to that of the white noise, shown in Fig.4 This is different to that shown in Fig.2 because of the extra filtering and attenuation in the analogue signal path. Most of the supersonic and subsonic frequencies are filtered out.

For more information on how IC1 produces white noise, see the *White Noise Generator* article published in the September 2019 issue.

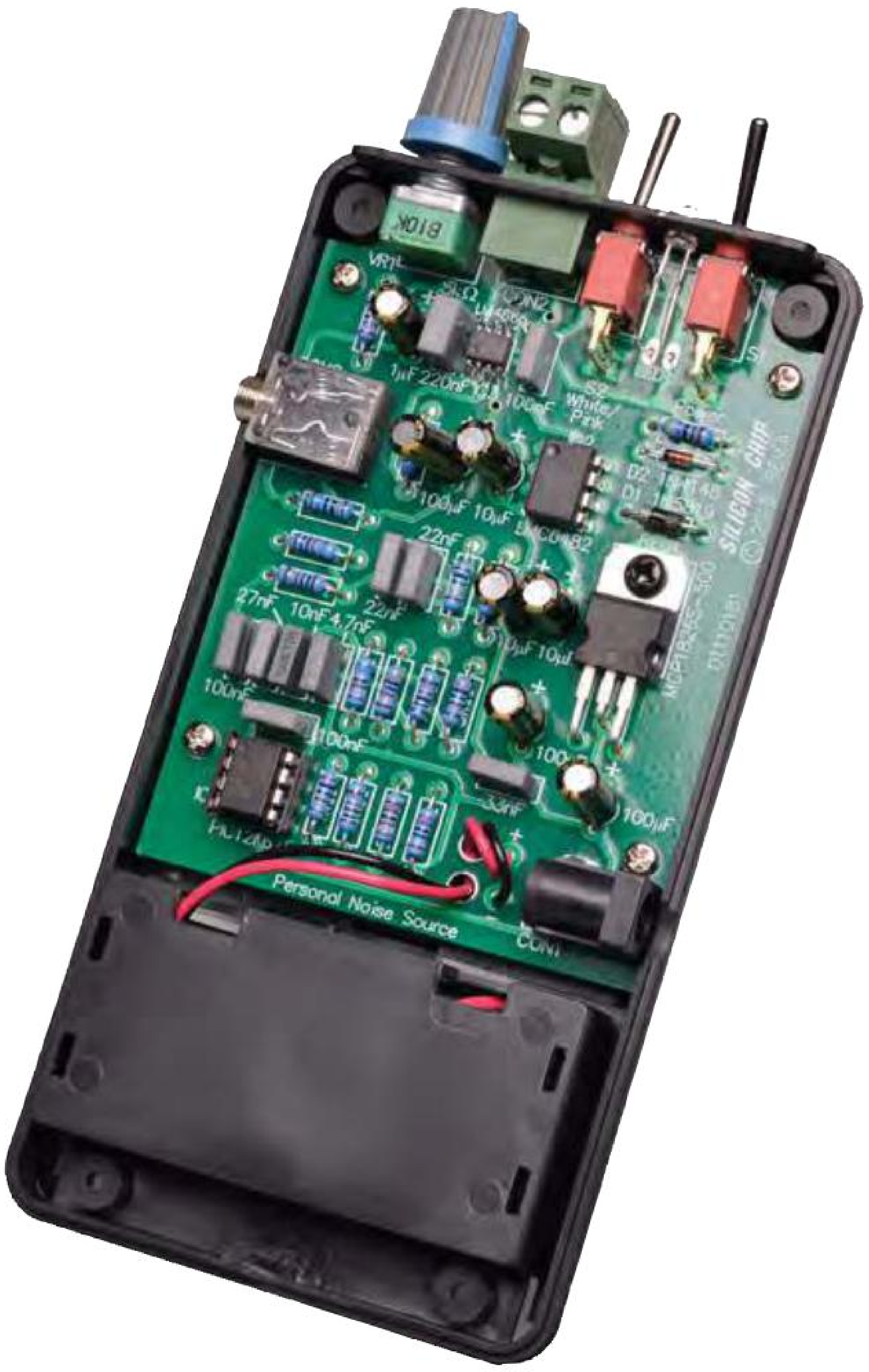
Filters

The white noise from IC1 is reduced in level using a resistive divider comprising 10kΩ and 270Ω resistors. This is so that the white noise is at a similar level to the pink noise, which means that switching between the two will not cause a noticeable jump in perceived volume. The supersonic (above 20kHz) signal components are then filtered out by a low-pass filter which consists of these two resistors plus a 33nF capacitor. The signal is then AC-coupled to non-inverting input pin 5 of buffer IC2b via a 22nF capacitor. This input pin is DC biased to half supply (around 2.5V) via the 1MΩ resistor, which connects to the junction of a voltage divider consisting of two 10kΩ resistors across the 5V supply. This half supply rail is decoupled to ground with a 10μF capacitor, so that supply noise is not injected into the signal via this path.

This DC biasing arrangement allows IC2b to produce a symmetrical swing within the 5V supply rail, and thus the amplifier output will not clip.

Pink noise filter

In the other signal path, the white noise signal becomes pink noise. It is first reduced in level by the 1kΩ and 2.2kΩ resistors. This reduction is not as great as that of the white noise signal path because the following filter


VR1
1µF 220nF
100nF
S2 White/ Pink
100µF 10µF
22nF
27nF 10nF 4.7nF
100nF
SILICON CHIP
100µF
Personal Noise Source
CON1

Here's how the PCB fits inside the case (in this instance it's the Jaycar case; the Altronics version actually mounts to the lid).

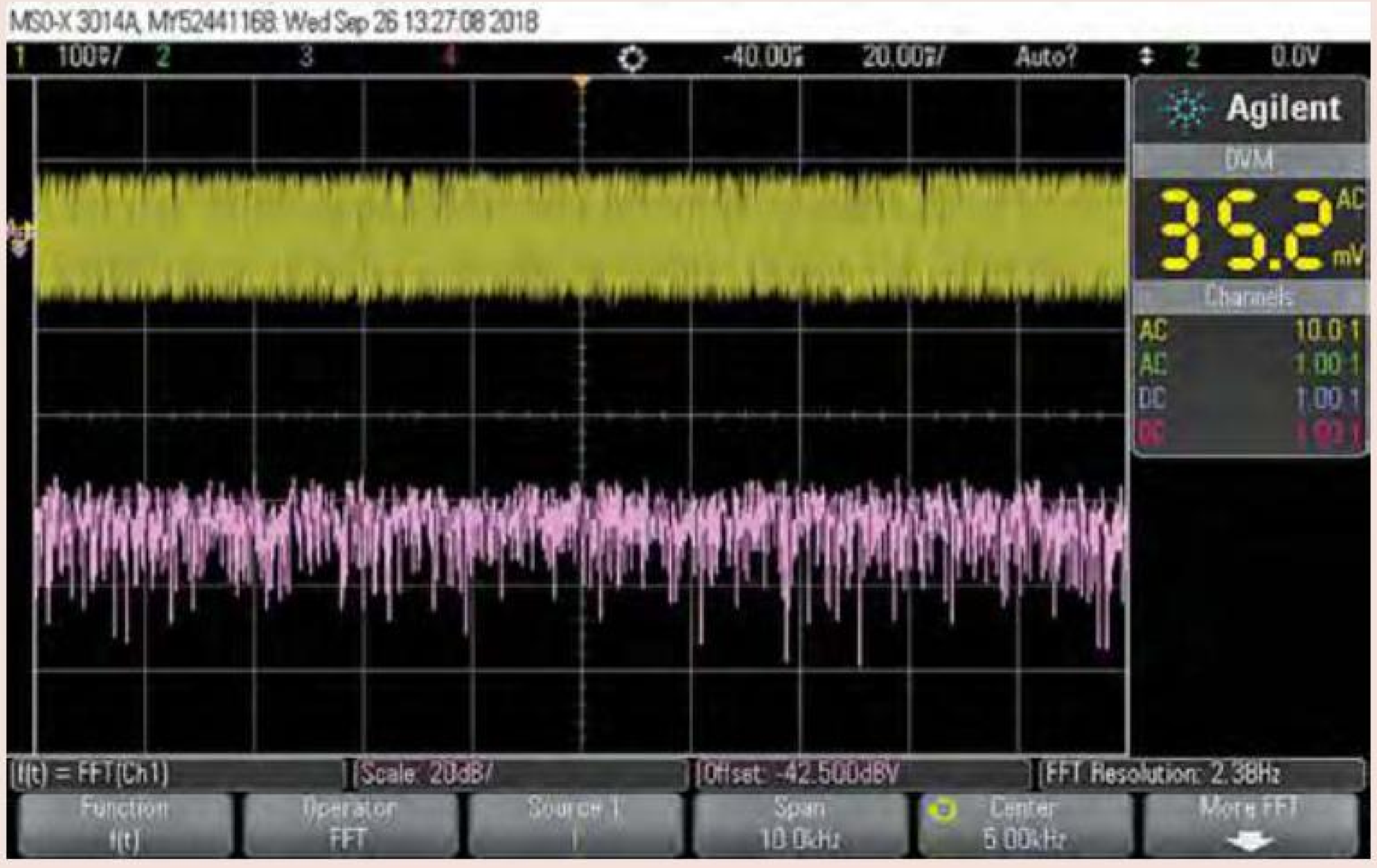

MSO-X 3014A, MY52441168: Wed Sep 26 13:27:08 2018
-40.00s
20.00s/
Auto?
0.0V
Agilent
DVM
35.2 mV AC
Channels
AC 10.0:1
AC 1.00:1
DC 1.00:1
DC 1.00:1
f(t) = FFT(Ch1)
Scale: 20dB/
Offset: -42.500dBV
FFT Resolution: 2.38Hz
Function f(t)
Operator FFT
Source 1
Span 10.0kHz
Center 5.00kHz
More FFT

Fig.2: the yellow trace shows the white noise output at pin 7 of IC1, with a spectrum analysis below showing the distribution of energy across various frequencies from a few hertz up to 10kHz. As you can see, its frequency distribution is effectively flat.

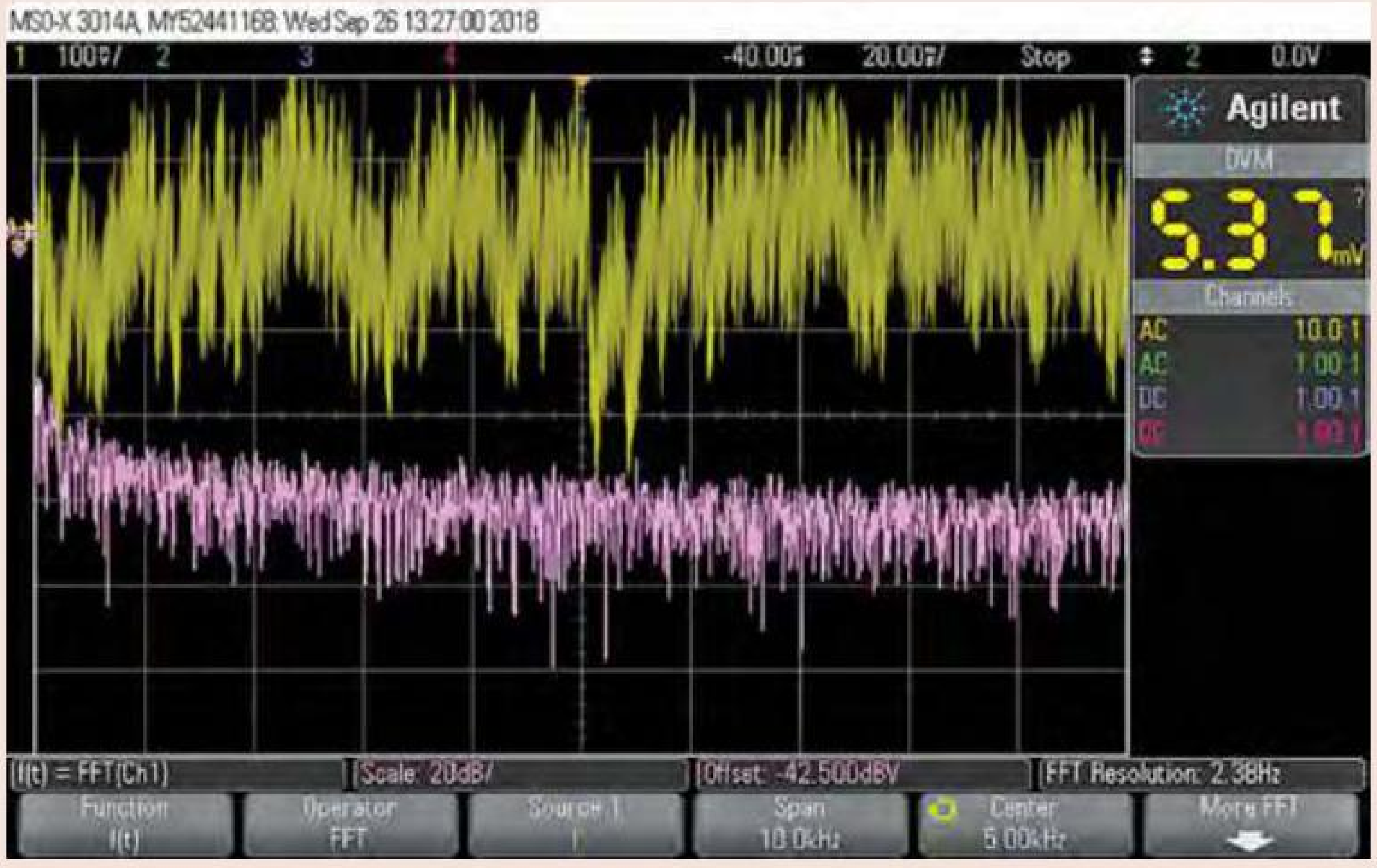

MSO-X 3014A, MY52441168: Wed Sep 26 13:27:00 2018
-40.00s
20.00s/
Stop
0.0V
Agilent
DVM
5.37 mV
Channels
AC 10.0:1
AC 1.00:1
DC 1.00:1
DC 1.00:1
f(t) = FFT(Ch1)
Scale: 20dB/
Offset: -42.500dBV
FFT Resolution: 2.38Hz
Function f(t)
Operator FFT
Source 1
Span 10.0kHz
Center 5.00kHz
More FFT

Fig.3: now the yellow trace shows the pink noise output at pin 1 of IC2a and the spectrum analysis below. We've 'zoomed in' to the 0-10kHz frequency range so you can see how the intensity falls off with increasing frequency in a logarithmic manner.

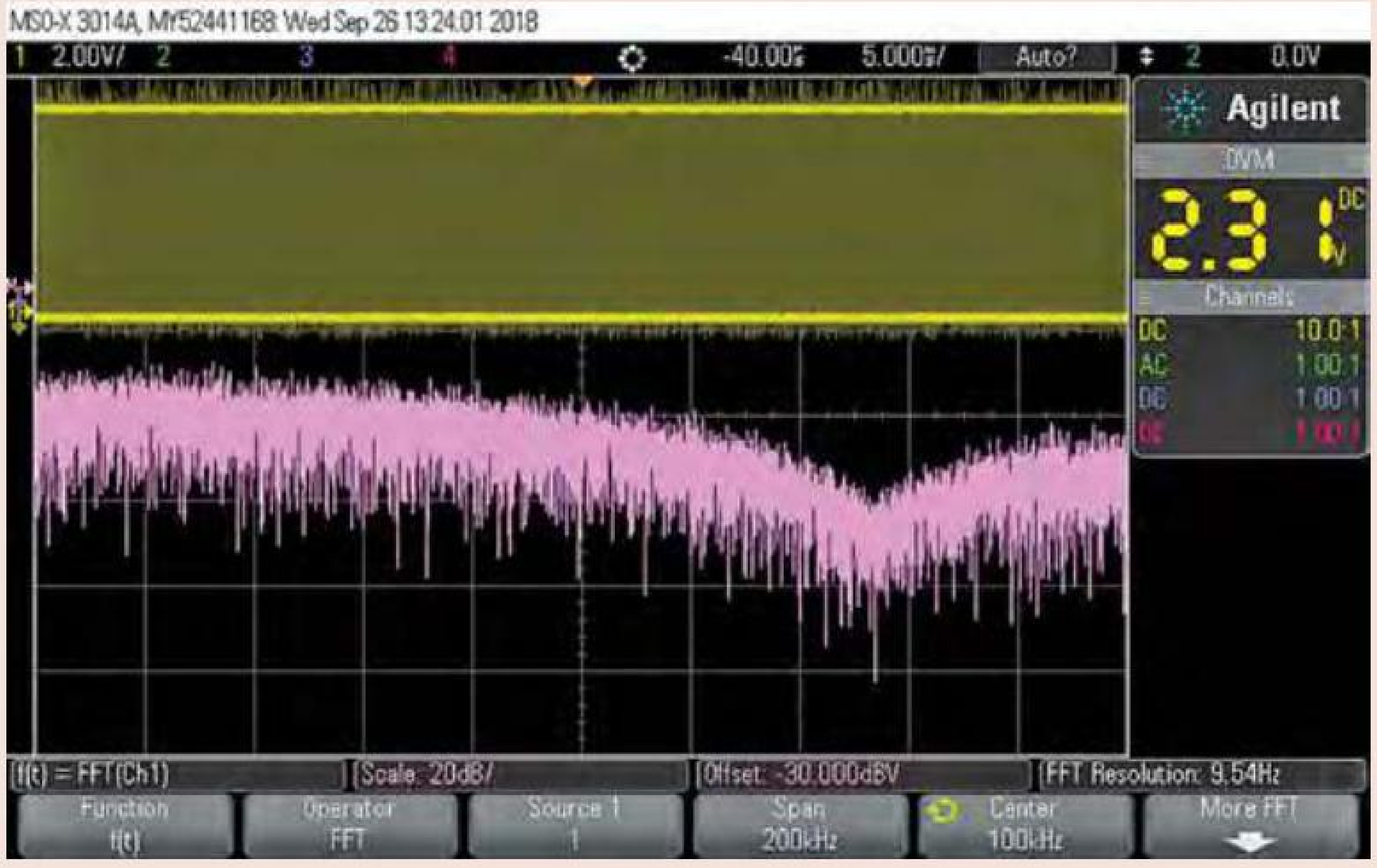

MSO-X 3014A, MY52441168: Wed Sep 26 13:24:01 2018
2.00V/
-40.00s
5.000s/
Auto?
0.0V
Agilent
DVM
2.31 V DC
Channels
DC 10.0:1
AC 1.00:1
DC 1.00:1
DC 1.00:1
f(t) = FFT(Ch1)
Scale: 20dB/
Offset: -30.000dBV
FFT Resolution: 9.54Hz
Function f(t)
Operator FFT
Source 1
Span 200kHz
Center 100kHz
More FFT

Fig.4: the raw white noise output at pin 7 of IC2b with spectrum analysis for 0-200kHz. Its amplitude is quite flat up to about 50kHz, rolling off to around –15dB at about 150kHz before increasing again, due to the harmonic content.

also provides some attenuation. The initial attenuation from these two resistors prevents clipping in the following buffer stage (IC2a).

The pink noise filter provides a –3dB per octave roll-off. That roll-off rate is difficult to achieve because an RC filter using a resistor and capacitor provides a higher roll-off rate, of 6dB per octave.

To get the –3dB per octave roll-off, a complex network of passive step filters is used. These combine to provide an overall response with the required roll-off rate.

This filter is based on one first published more than 40 years ago (1976) in the *National Semiconductor Audio Handbook* (see page 2-56 of **http://bit.ly/pe-nov19-nah**).

This filter is accurate to within ±0.25dB from 10Hz to 40kHz when close-tolerance components are used. The resulting signal is AC-coupled via a 22nF capacitor to the non-inverting input of buffer IC2a and biased with a 1MΩ resistor using the same arrangement as for IC2b.

The selected signal (white or pink noise) at the common terminal of switch S2 is applied to the input of an LM4865 audio amplifier (IC3) via a 220nF AC-coupling capacitor.

Amplifier operation

When headphones are connected, IC3 drives them via a 100µF electrolytic coupling capacitor from output pin 5. The capacitor removes any DC bias from the amplifier's output.

The headphone socket (CON3) tip and ring connections are joined together so that both sides of the headphones/earphones are driven in parallel.

A 150Ω resistor ensures that the headphone side of this capacitor is DC-biased to ground even if the headphones are not plugged in, so that when they are plugged in, there isn't a loud thump as the capacitor charges.

When headphones are not used, IC3 will instead drive a loudspeaker in a bridge-tied-load (BTL) arrangement. The BTL configuration means that when the voltage at the pin 8 output goes positive, the voltage at the pin 5 output goes negative and vice versa.

This results in twice the voltage across the speaker compared to that at either output pin, giving up to four times the output power ($V^2 \times R$). It also eliminates the need for a coupling capacitor, since both ends of the speaker are driven with the same DC bias level.

When headphones are plugged into CON3, the internal switch is open and so the HP-sense input (pin 3) of IC3 is pulled to +5V by the connected 100kΩ resistor. This shuts down the pin 8 output, making it high impedance and thus muting any speaker connected via CON2.

It also reduces IC3's power consumption when driving headphones, since the second internal amplifier is also shut down and not drawing any current.

With headphones not plugged in, the switch contact closes and the 150Ω resistor pulls pin 3 below 50mV. This enables the BTL drive arrangement for the speaker.

IC3 has a DC volume control input at pin 4. Potentiometer VR1 is used to adjust the voltage at this pin between 0V, for minimum volume, through to about 4.1V for maximum volume, when wound fully clockwise. The 4.1V maximum level is achieved using a 2.2kΩ padding resistor between VR1 and the +5V supply.

Power supply

Both IC1 and IC3 need a 5V supply, so the entire circuit is powered from the 5V supply rail. This is provided by linear regulator REG1, which is fed by 9V from the battery or 6-12V DC from a plugpack connected via CON1.

REG1 has a low quiescent current and a low dropout voltage, meaning it won't drain the battery too fast and it can provide a steady 5V output even when the battery voltage is barely above 5V.

Schottky diode D1 provides protection in case either supply is connected with incorrect polarity. Also, the switch within DC barrel socket (CON1) disconnects the battery when a DC plug is inserted.

The unit is powered on or off using toggle switch S1. Blue LED1 lights up when it is on. This LED has a 3V voltage drop and diode D2, which is connected in series, has a forward voltage of around 0.7V. This means that the LED will only light if the regulator output is above about 3.7V.

In fact, the LED will be very dim unless the supply is close to 5V. So LED1 is not only a power indicator, but also works as a battery voltage indicator, showing when REG1 drops out of regulation as the battery becomes discharged.

Construction

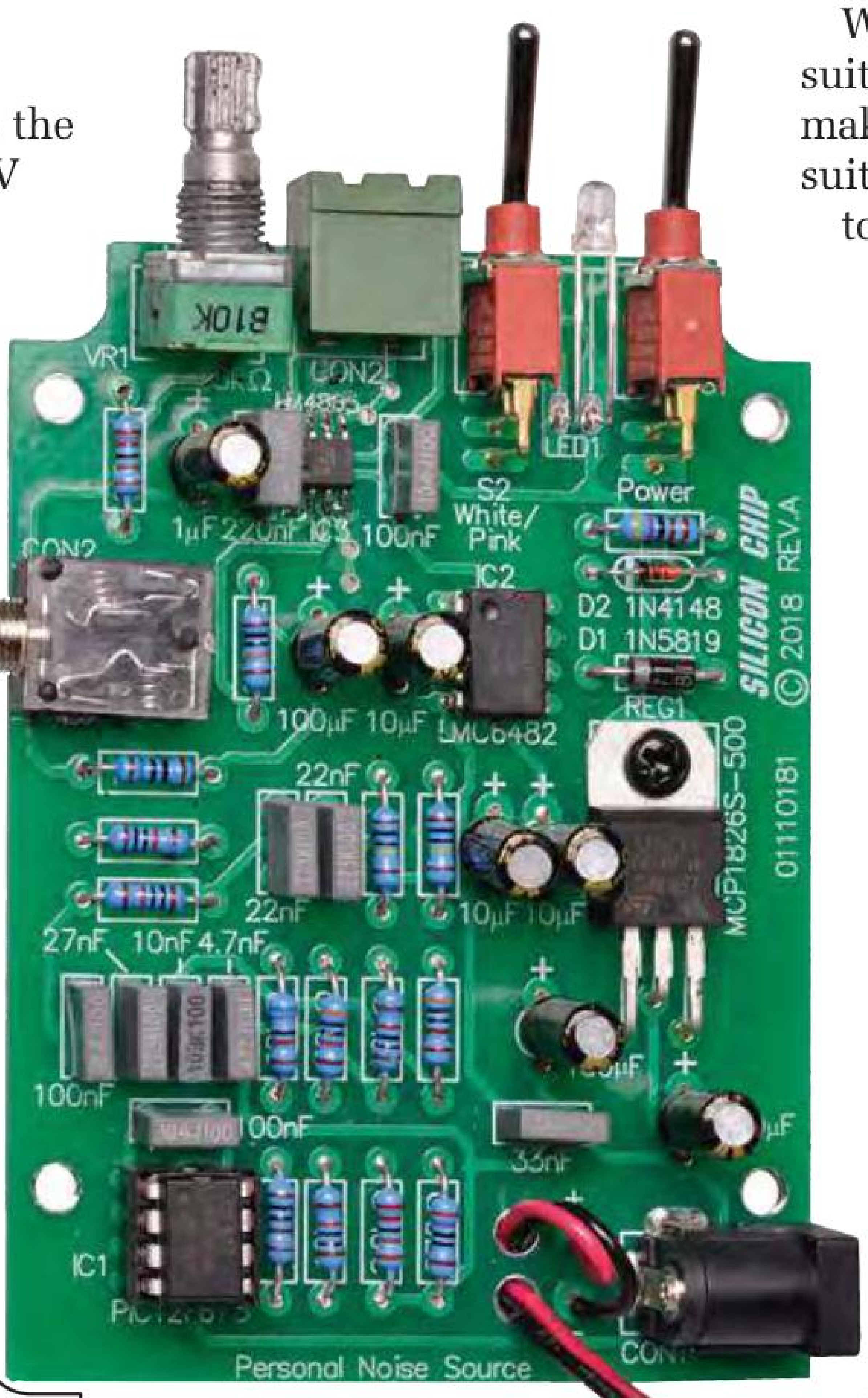
The *Tinnitus & Insomnia Killer* is designed to be housed in one of two similar (but slightly different size) cases. There are two different PCB designs. One is coded 01110181 and measures 63.5mm x 86mm, which suits the Jaycar HB5610 case (135 x 70 x 24mm). The other is coded 01110182 and measures 58.5 x 86mm; this one suits the Altronics H0342 case (130 x 68 x 25mm). Both are avaiable from the *PE PCB Service* – see page 76.

We have also produced panel labels to suit both boxes. Before starting assembly, make sure you have the correct PCB to suit your chosen case. They are shaped to fit inside the respective case and mount on the integral plastic posts.

Use the appropriate PCB overlay diagram, either Fig.5 (01110181) or Fig.6 (01110182) and the matching photo (built into the Jaycar case) as a guide during assembly.

Start by fitting surface-mount IC3. Solder it directly to the PCB. First, check the overlay diagram for the correct orientation, then tack solder one pin to the board.

Some constructors find that using a wooden clothes peg (not plastic – it melts!) helps to hold



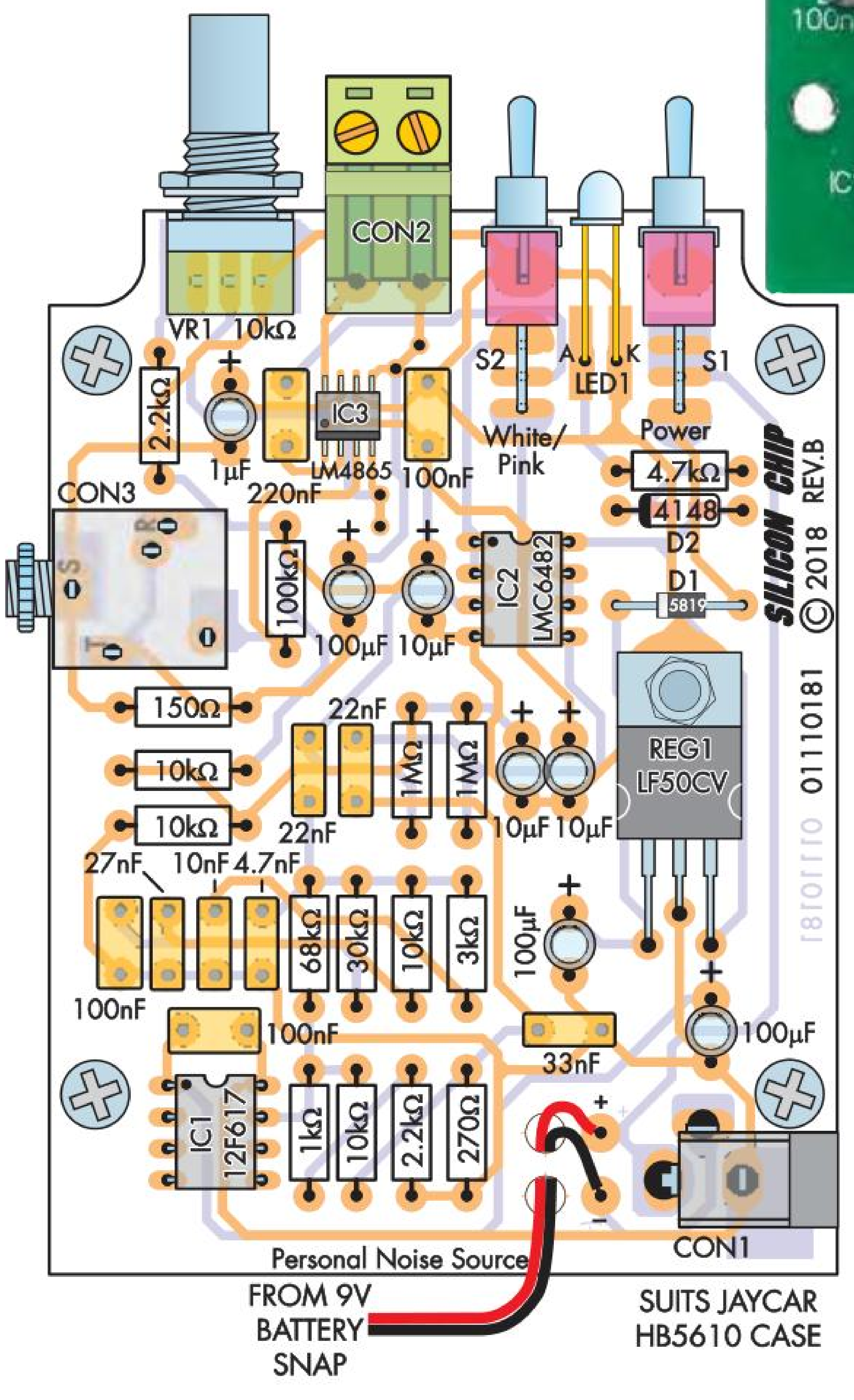

VR1 10kΩ
CON2
S2
A K
LED1
S1
2.2kΩ
IC3
LM4865
1µF
220nF
100nF
White/
Pink
Power
4.7kΩ
4148
D2
D1
5819
CON3
100kΩ
IC2
LMC6482
100µF 10µF
SILICON CHIP
© 2018 REV.B
01110181
150Ω
22nF
1MΩ
1MΩ
REG1
LF50CV
10kΩ
10kΩ
22nF
10µF 10µF
27nF 10nF 4.7nF
68kΩ
30kΩ
10kΩ
3kΩ
100µF
100nF
100nF
33nF
100µF
IC1
12F617
1kΩ
10kΩ
2.2kΩ
270Ω
Personal Noise Source
CON1
FROM 9V
BATTERY
SNAP
SUITS JAYCAR
HB5610 CASE

Fig.5 (left): use this PCB overlay as a guide when building the board that fits the Jaycar case. Be careful with the polarity of D1, D2, ICs1-3 and the electrolytic capacitors. Make sure these capacitors sit low on the board before soldering them or they may not fit the case.

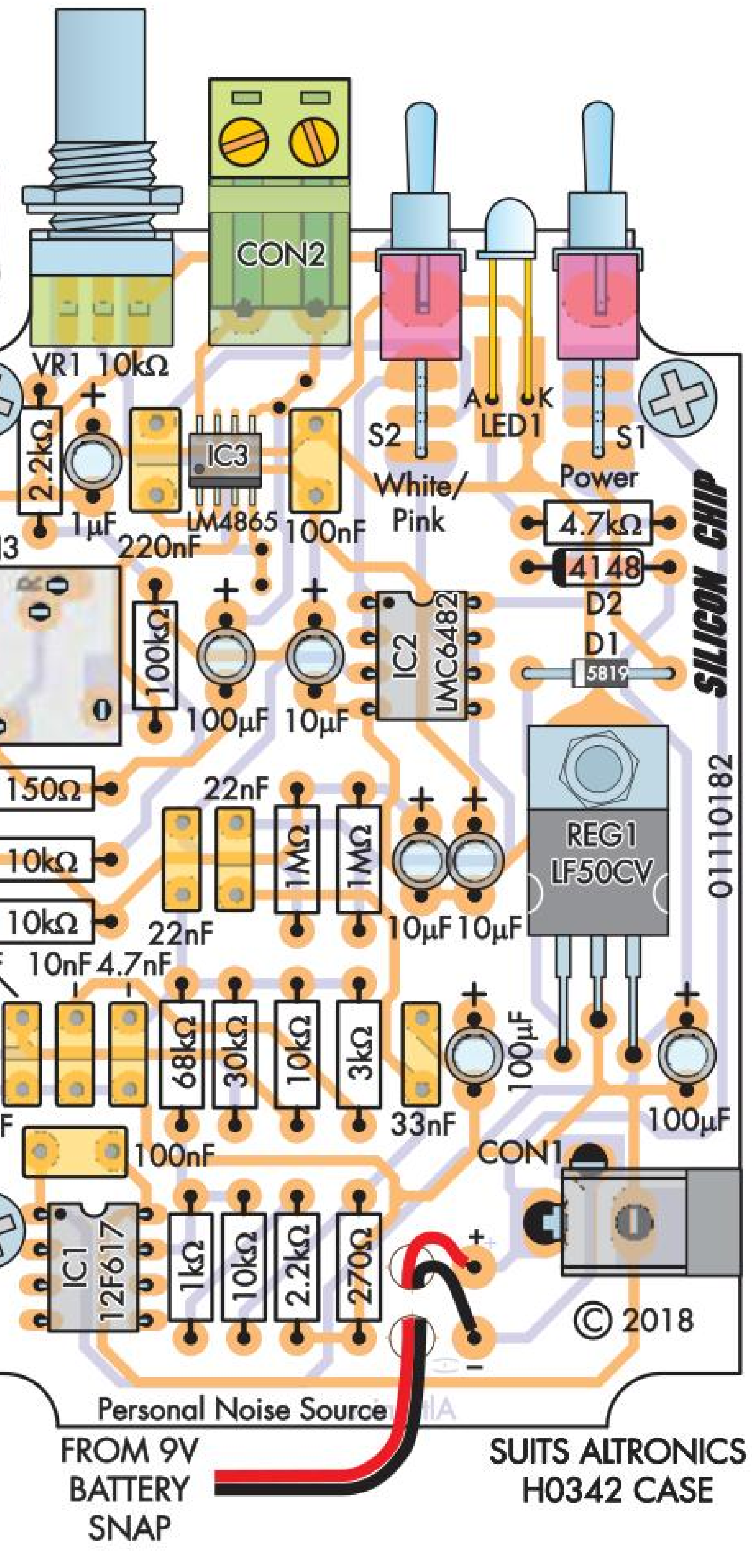

VR1 10kΩ
CON2
S2
A K
LED1
S1
2.2kΩ
IC3
LM4865
1µF
220nF
100nF
White/
Pink
Power
4.7kΩ
4148
D2
D1
5819
CON3
100kΩ
IC2
LMC6482
100µF 10µF
SILICON CHIP
01110182
150Ω
22nF
1MΩ
1MΩ
REG1
LF50CV
10kΩ
10kΩ
22nF
10µF 10µF
27nF 10nF 4.7nF
68kΩ
30kΩ
10kΩ
3kΩ
33nF
100µF
100µF
100nF
100nF
CON1
IC1
12F617
1kΩ
10kΩ
2.2kΩ
270Ω
© 2018
Personal Noise Source
FROM 9V
BATTERY
SNAP
SUITS ALTRONICS
H0342 CASE

Fig.6 (right): PCB overlay for the board that fits the Altronics case, which is slightly narrower and has different mounting hole locations. The component arrangement and interconnections are otherwise identical. Be sure to do up REG1's screw before soldering its leads to prevent damage.

Parts list – Tinnitus/Insomnia Killer

1 double-sided PCB coded 01110181 (63.5mm x 86mm) [to suit Jaycar case] ***or***
1 double-sided PCB coded 01110182 (58.5 x 86mm) [to suit Altronics case]
1 remote control case, 135 x 70 x 24mm [Jaycar HB5610]
or
1 remote control case, 130 x 68 x 25mm [Altronics H0342]
and
1 remote control belt clip [Altronics H0349] (optional)
1 front panel label, to suit case
1 8-pin DIL socket (for IC1)
2 SPDT PCB-mount toggle switches [Altronics S1421]
1 9V battery and battery snap
1 2.1mm or 2.5mm ID switched DC socket (CON1) [Jaycar PS0519, Altronics P0620 or P0621A]
1 2-way right-angle pluggable terminal block socket (CON2) [Altronics P2592, Jaycar HM3102]
1 2-way pluggable screw terminal block (CON2) [Altronics P2512, Jaycar HM3122]
1 3.5mm PCB-mount stereo jack socket (CON2) [Jaycar PS0133, Altronics P0092]
1 M3 x 6mm screw and M3 hex nut (for mounting REG1)
4 No.4 self-tapping screws (for Jaycar case only)
1 knob to suit VR1

Semiconductors
1 PIC12F617-I/P microcontroller programmed with 0111018A.HEX (IC1)
1 LMC6482AIN dual rail-to-rail op amp (IC2)
1 LM4865M audio power amplifier, SOIC-8 (IC3)
1 LF50CV low dropout, low quiescent current 5V regulator (REG1) [element14 Cat 1094240]
1 3mm blue LED (LED1)
1 1N5819 1A schottky diode (D1)
1 1N4148 signal diode (D2)

Capacitors
3 100µF 16V PC electrolytic
3 10µF 16V PC electrolytic
1 1µF 16V PC electrolytic
1 220nF MKT polyester
3 100nF MKT polyester
1 33nF MKT polyester
1 27nF MKT polyester (***or*** 22nF and 4.7nF in parallel)
2 22nF MKT polyester
1 10nF MKT polyester
1 4.7nF MKT polyester

Resistors (all 0.25W, 1% metal film)
2 1MΩ 1 100kΩ 1 68kΩ 1 30kΩ 4 10kΩ
1 4.7kΩ 1 3kΩ 2 2.2kΩ 1 1kΩ 1 270Ω 1 150Ω
1 9mm 10kΩ linear potentiometer (VR1) [Jaycar RP8510]

small SMD components in place while soldering the first pin. Now re-check the orientation and that all the pins are positioned correctly over their pads before soldering the remaining pins. If it is misaligned, remelt the solder on the first pin and adjust its position. Any solder bridges between the leads can be cleared by adding a small amount of flux paste and then using solder wick to draw up excess solder.

Next, mount the resistors; we recommend that you measure each value using a digital multimeter before fitting them.

You can then install the diodes. These must be mounted with the orientation shown. D1 is a 1N5819 type, while D2 is a smaller 1N4148 type.

IC1 should preferably be mounted in an IC socket, while IC2 can be soldered directly to the PCB. When installing the socket and ICs, take care to orient them correctly. The small dimple marking pin 1 must be positioned as shown in the relevant overlay diagram.

REG1 mounts horizontally on the PCB with the leads bent down 90° to insert into the holes. The metal tab is secured to the PCB with an M3 screw and nut.

Make sure you bend the pins down and tighten the screw *before* soldering the leads; otherwise, when you do it up, you could crack the solder joints.

The capacitors can be mounted next, starting with the MKT types. There are two options for the 27nF capacitor, as mentioned in the parts list. It's easiest to use a single 27nF capacitor but if you can't get one, you can solder a 22nF capacitor in its place on the top of the PCB and add a 4.7nF capacitor mounted on its side under the PCB (so they're soldered in parallel).

The electrolytic types should go in next. Once again, they must be oriented with the polarity shown; ie, with the longer (positive) lead through the hole nearest the '+' symbol on the PCB. The stripe on the can indicates the negative lead.

Make sure these capacitors are mounted hard down in the PCB surface, so their height above the PCB is no more than 12.5mm; otherwise, the lid of the case will not fit correctly.

The potentiometer (VR1) and PCB-mounted switches S1 and S2 can now be fitted, along with the DC socket (CON1), the terminals for the loudspeaker (CON2) and the 3.5mm jack socket (CON3).

Finally, solder LED1 in place. It's mounted with its lens horizontal, centred at a height of 6mm above the PCB. Bend its leads at 14mm back from the base of the lens through 90°, making sure the longer anode lead is to the left.

Testing

Apply power (either from a 9V battery or plugpack) and check that LED1 lights and that REG1 provides a 5V output, measured between its metal tab and the right-hand lead (nearest the edge of the PCB).

Also, check for 5V at pin 1 of IC1, pin 8 of IC2 and pin 1 of IC3. Pins 3 and 5 of IC2 should be at around 2.5V.

Turn volume control VR1 down to zero (maximum anticlockwise) then plug in a pair of headphones or earbuds. Put them on – you should hear nothing – then slowly turn VR1 up and check that you can hear the sound output.

Unplug the headphones and repeat the above check with an external speaker connected to CON2 now. You should be rewarded with an increase in noise as you increase VR1.

For both earphones or speakers, pink noise is produced when switch S2 is in the up position and white noise when it is down.

Preparing the case

Because all the controls and sockets are mounted directly on the PCB, it is essential that they are drilled/cut out in the proper position.

Use Fig.7 as a guide for locating and sizing these holes. You can download this diagram as a PDF file from the November 2019 page of the *PE* website, print it and use it as a template.

Most holes can simply be drilled (with care) but the 12.5 × 9mm rectangular hole (for the speaker terminal block) is made by drilling a circular hole or series of holes within the perimeter and then filing to shape.

Holes are also required in the sides of the case for the DC socket and 3.5mm jack socket. Place the PCB in the case and mark out the positions for the holes required. The Jaycar case has a removable end panel which makes drilling a little easier.

The Altronics case does not have such a panel – in this case the PCB mounts upside-down on the lid. Therefore,

Small Capacitor Codes

	Qty.	Value	µF Code	EIA Code	IEC Code
☐	1	220nF	0.22µF	224	220n
☐	3	100nF	0.1µF	104	100n
☐	1	33nF	0.033µF	333	33n
☐	1	27nF	0.027µF	273	27n
☐	2	22nF	0.022µF	223	22n
☐	1	10nF	0.01µF	103	10n
☐	1	4.7nF	0.0047µF	472	4n7

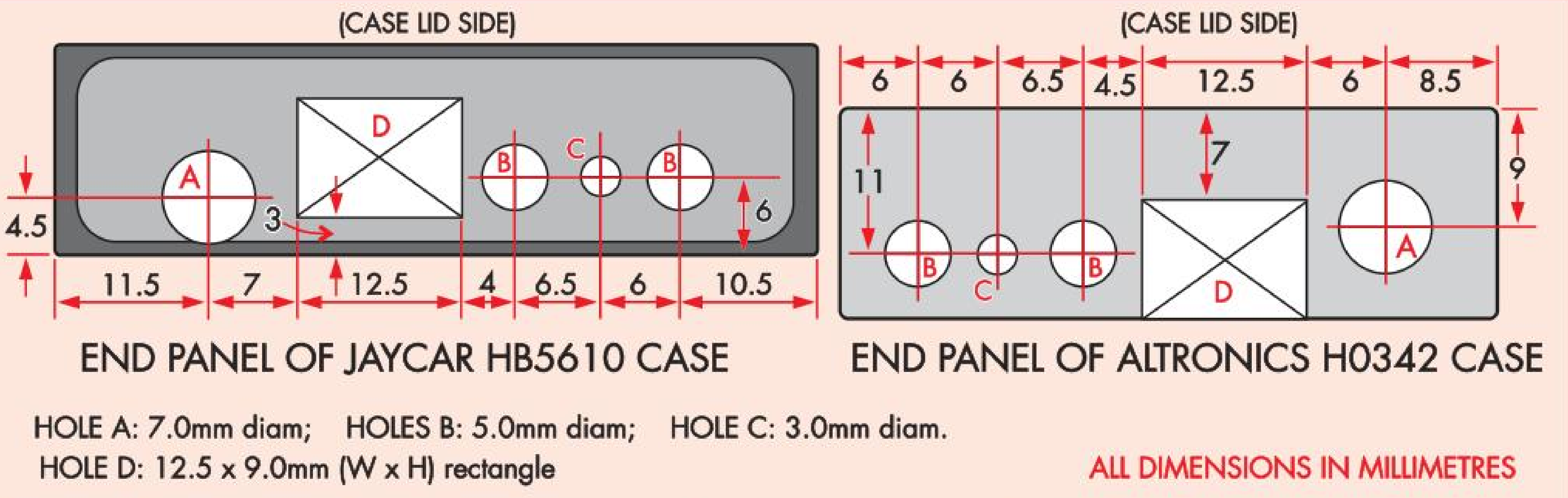

(CASE LID SIDE)
(CASE LID SIDE)
A
B
C
D
4.5
3
6
11.5
7
12.5
4
6.5
6
10.5
6
6
6.5
4.5
12.5
6
8.5
11
7
9
END PANEL OF JAYCAR HB5610 CASE
END PANEL OF ALTRONICS H0342 CASE
HOLE A: 7.0mm diam; HOLES B: 5.0mm diam; HOLE C: 3.0mm diam.
HOLE D: 12.5 x 9.0mm (W x H) rectangle
ALL DIMENSIONS IN MILLIMETRES

Fig.7: drilling and cutting patterns for the end panels of the two cases. The reason they are different is that the Jaycar PCB is mounted normally in the case, whereas the Altronics PCB is 'hung' upside down from its case lid (which becomes the front panel). The rectangular cut-outs can be made by drilling a series of small holes around the outside, then carefully filing the hole to shape.

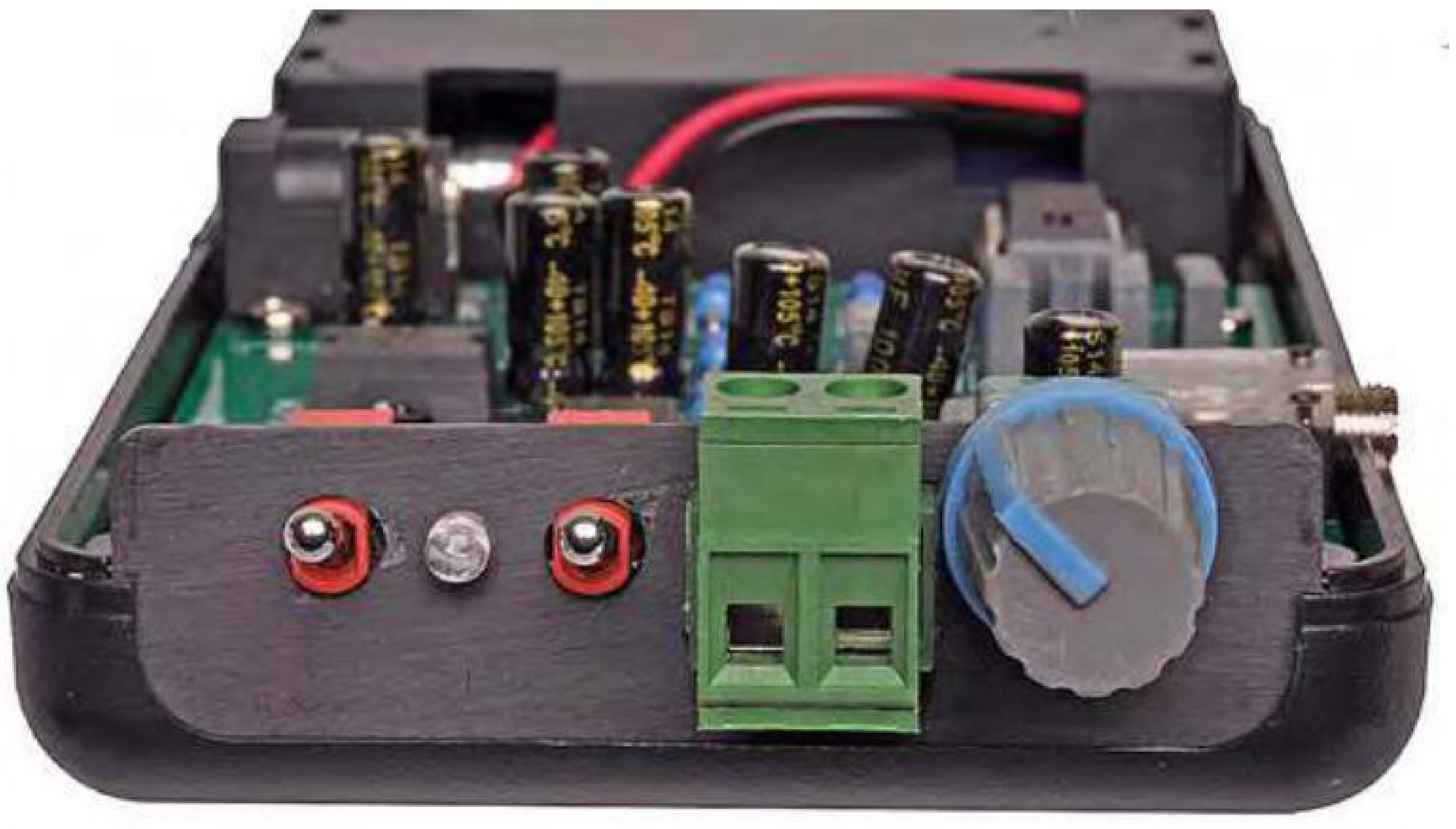
the holes are in a different position to those in the Jaycar case – don't get cases and hole mixed up!

Also, you will need to remove the retaining clip from the plug for CON2 with side cutters, as this will foul the panel and case if left intact, preventing its insertion.

Final assembly

For the Jaycar case, the battery snap is inserted from the battery compartment side and the leads pass through to the PCB. They are routed through two 3mm holes for strain relief, as shown in Figs. 5 and 6.

Solder the ends directly to the plus and minus pads, ensuring that the red lead goes to the pad marked 'plus'. The PCB is secured to the base of the case using four self-tapping screws for the Jaycar version and using three screws for the Altronics version, into the integral mounting bushes.



End-on view of the controls of the *Tinnitus & Insomnia Killer* – again, this is the Jaycar case version. No label is needed as markings are on the front panel .

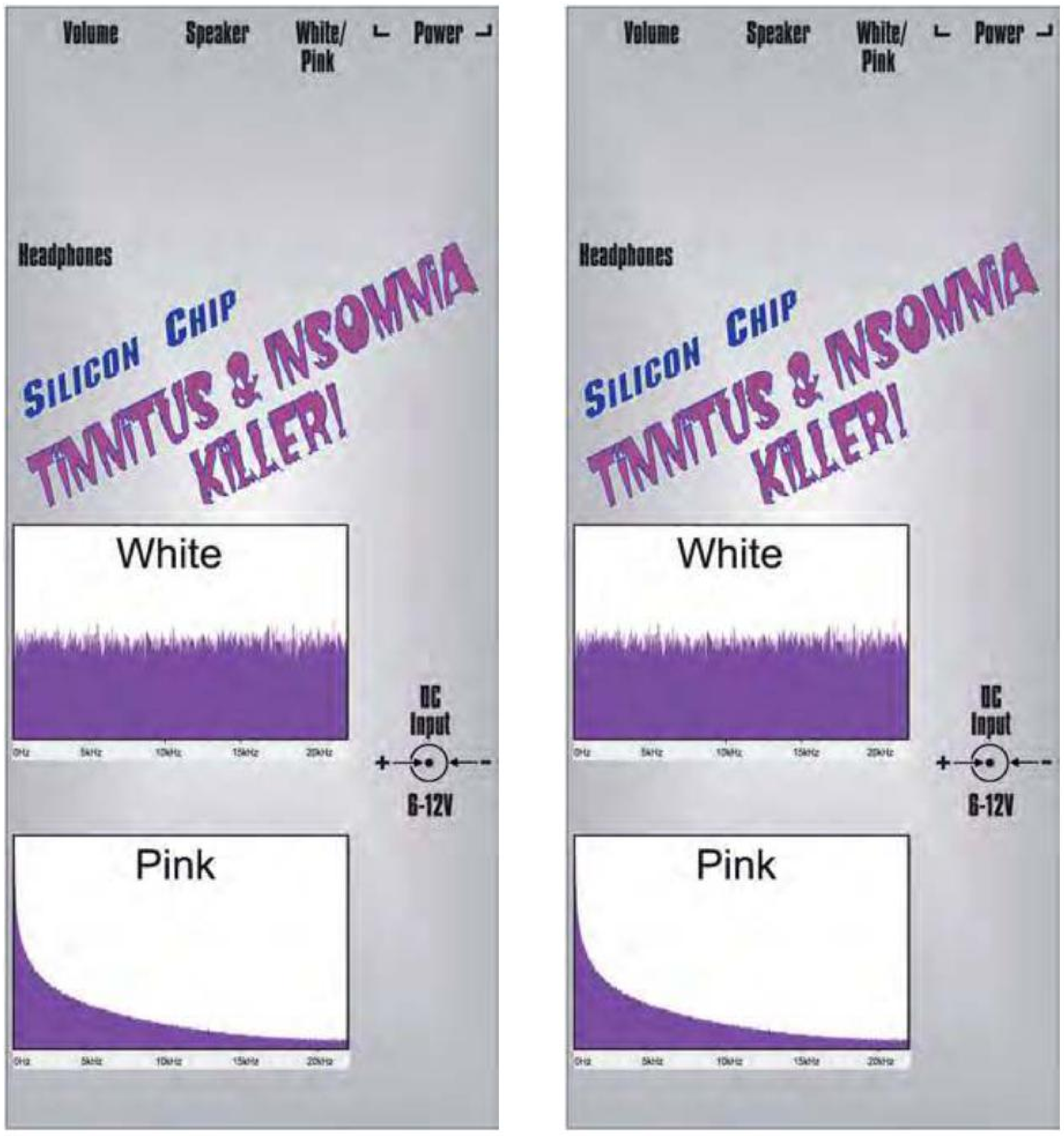

Volume
Speaker
White/ Pink
Power
Headphones
SILICON CHIP
TINNITUS & INSOMNIA KILLER!
White
Pink
DC Input
6-12V
Volume
Speaker
White/ Pink
Power
Headphones
SILICON CHIP
TINNITUS & INSOMNIA KILLER!
White
Pink
DC Input
6-12V

Fig.8: full-size front panel artwork for the Jaycar case (left) and the Altronics case (right). They are slightly different sizes to match the different case sizes. These can be downloaded from the November 2019 page of the *PE* website.

If you purchased the optional belt clip for the Altronics case, attach it now, then attach the lid to the case using the four screws supplied with the case.

Front panel label

To produce a front panel label, download it from the November 2019 page of the *PE* website. For more information on making labels, see **siliconchip.com.au/Help/FrontPanels**

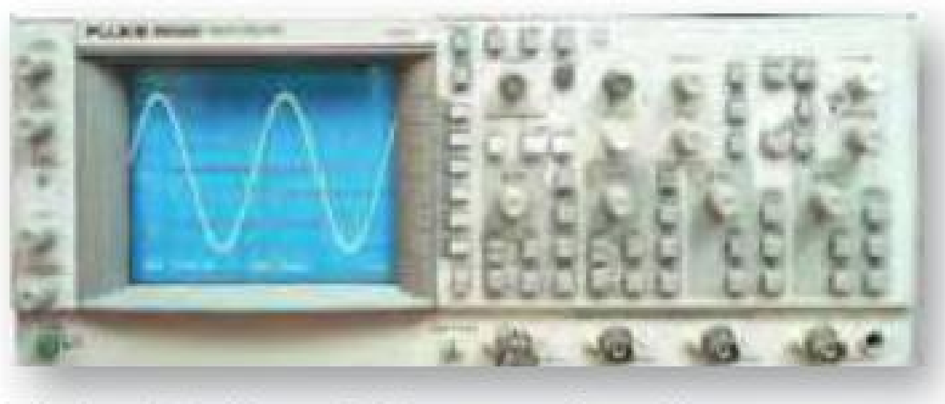
Which speaker to use?

Just about any 4 or 8Ω speaker can be pressed into service. Maximum power is only 750mW, so you're not likely to blow anything up!

And contrary to popular belief, larger speakers generally do not require more power to drive than smaller speakers, as they are (usually) more efficient.

Therefore, the larger one will usually sound 'louder' than a smaller one for a given power input. So if you want to use that old speaker box gathering dust in the cupboard, go ahead and hook it up!

Reproduced by arrangement with
SILICON CHIP magazine 2019.
www.siliconchip.com.au


Fluke/Philips PM3092 Oscilloscope
2+2 Channel 200MHz Delay TB,
Autoset etc – £250

STEWART OF READING
17A King Street, Mortimer, near Reading, RG7 3RS
Telephone: **0118 933 1111** Fax: **0118 933 2375**
USED ELECTRONIC TEST EQUIPMENT
Check website **www.stewart-of-reading.co.uk**
(ALL PRICES PLUS CARRIAGE & VAT)
Please check availability before ordering or calling in

HP 54600B Oscilloscope
Analogue/Digital Dual Trace 100MHz
Only £75, with accessories £125

LAMBDA GENESYS	PSU GEN100-15 100V 15A Boxed As New	£400
LAMBDA GENESYS	PSU GEN50-30 50V 30A	£400
IFR 2025	Signal Generator 9kHz – 2.51GHz Opt 04/11	£900
IFR 2948B	Communication Service Monitor Opts 03/25 Avionics	POA
IFR 6843	Microwave Systems Analyser 10MHz – 20GHz	POA
R&S APN62	Syn Function Generator 1Hz – 260kHz	£295
Agilent 8712ET	RF Network Analyser 300kHz – 1300MHz	POA
HP8903A/B	Audio Analyser	£750 – £950
HP8757D	Scaler Network Analyser	POA
HP3325A	Synthesised Function Generator	£195
HP3561A	Dynamic Signal Analyser	£650
HP6032A	PSU 0-60V 0-50A 1000W	£750
HP6622A	PSU 0-20V 4A Twice or 0-50V 2A Twice	£350
HP6624A	PSU 4 Outputs	£400
HP6632B	PSU 0-20V 0-5A	£195
HP6644A	PSU 0-60V 3.5A	£400
HP6654A	PSU 0-60V 0-9A	£500
HP8341A	Synthesised Sweep Generator 10MHz – 20GHz	£2,000
HP83630A	Synthesised Sweeper 10MHz – 26.5 GHz	POA
HP83624A	Synthesised Sweeper 2 – 20GHz	POA
HP8484A	Power Sensor 0.01-18GHz 3nW-10µW	£75
HP8560E	Spectrum Analyser Synthesised 30Hz – 2.9GHz	£1,750
HP8563A	Spectrum Analyser Synthesised 9kHz – 22GHz	£2,250
HP8566B	Spectrum Analsyer 100Hz – 22GHz	£1,200
HP8662A	RF Generator 10kHz – 1280MHz	£750
Marconi 2022E	Synthesised AM/FM Signal Generator 10kHz – 1.01GHz	£325
Marconi 2024	Synthesised Signal Generator 9kHz – 2.4GHz	£800
Marconi 2030	Synthesised Signal Generator 10kHz – 1.35GHz	£750
Marconi 2023A	Signal Generator 9kHz – 1.2GHz	£700
Marconi 2305	Modulation Meter	£250
Marconi 2440	Counter 20GHz	£295
Marconi 2945/A/B	Communications Test Set Various Options	POA
Marconi 2955	Radio Communications Test Set	£595
Marconi 2955A	Radio Communications Test Set	£725
Marconi 2955B	Radio Communications Test Set	£800
Marconi 6200	Microwave Test Set	£1,500
Marconi 6200A	Microwave Test Set 10MHz – 20GHz	£1,950
Marconi 6200B	Microwave Test Set	£2,300
Marconi 6960B	Power Meter with 6910 sensor	£295
Tektronix TDS3052B	Oscilloscope 500MHz 2.5GS/s	£1,250
Tektronix TDS3032	Oscilloscope 300MHz 2.5GS/s	£995
Tektronix TDS3012	Oscilloscope 2 Channel 100MHz 1.25GS/s	£450
Tektronix 2430A	Oscilloscope Dual Trace 150MHz 100MS/s	£350
Tektronix 2465B	Oscilloscope 4 Channel 400MHz	£600
Farnell AP60/50	PSU 0-60V 0-50A 1kW Switch Mode	£300
Farnell XA35/2T	PSU 0-35V 0-2A Twice Digital	£75
Farnell AP100-90	Power Supply 100V 90A	£900
Farnell LF1	Sine/Sq Oscillator 10Hz – 1MHz	£45
Racal 1991	Counter/Timer 160MHz 9 Digit	£150
Racal 2101	Counter 20GHz LED	£295
Racal 9300	True RMS Millivoltmeter 5Hz – 20MHz etc	£45
Racal 9300B	As 9300	£75
Solartron 7150/PLUS	6½ Digit DMM True RMS IEEE	£65/£75
Solatron 1253	Gain Phase Analyser 1mHz – 20kHz	£600
Solartron SI 1255	HF Frequency Response Analyser	POA
Tasakago TM035-2	PSU 0-35V 0-2A 2 Meters	£30
Thurlby PL320QMD	PSU 0-30V 0-2A Twice	£160 – £200
Thurlby TG210	Function Generator 0.002-2MHz TTL etc Kenwood Badged	£65

HP/Agilent HP 34401A Digital
Multimeter 6½ Digit £325 – £375

HP33120A	Function Generator 100 microHz – 15MHz	£350
HP53131A	Universal Counter 3GHz Boxed unused	£600
HP53131A	Universal Counter 225MHz	£350
Audio Precision	SYS2712 Audio Analyser – in original box	POA
Datron 4708	Autocal Multifunction Standard	POA
Druck DPI 515	Pressure Calibrator/Controller	£400
Datron 1081	Autocal Standards Multimeter	POA
ENI 325LA	RF Power Amplifier 250kHz – 150MHz 25W 50dB	POA
Keithley 228	Voltage/Current Source	POA
Time 9818	DC Current & Voltage Calibrator	POA

Marconi 2955B Radio
Communications Test Set – £800

CRICKLEWOOD ELECTRONICS
Established 1981

Your best bet since MAPLIN
Chock-a-Block with Stock

Visit: www.cricklewoodelectronics.com
Or phone our friendly knowledgeable staff on **020 8452 0161**
Components • Audio • Video • Connectors • Cables
Arduino • Test Equipment etc, etc

Visit our Shop, Call or Buy online at:
www.cricklewoodelectronics.com
020 8452 0161
Visit our shop at:
40-42 Cricklewood Broadway
London NW2 3ET

ESR Electronic Components Ltd

All of our stock is RoHS compliant and CE approved. Visit our well stocked shop for all of your requirements or order on-line.

We can help and advise with your enquiry, from design to construction.

3D Printing • Cable • CCTV • Connectors • Components • Enclosures • Fans • Fuses • Hardware • Lamps • LED's • Leads • Loudspeakers • Panel Meters • PCB Production • Power Supplies • Relays • Resistors • Semiconductors • Soldering Irons • Switches • Test Equipment • Transformers and so much more…

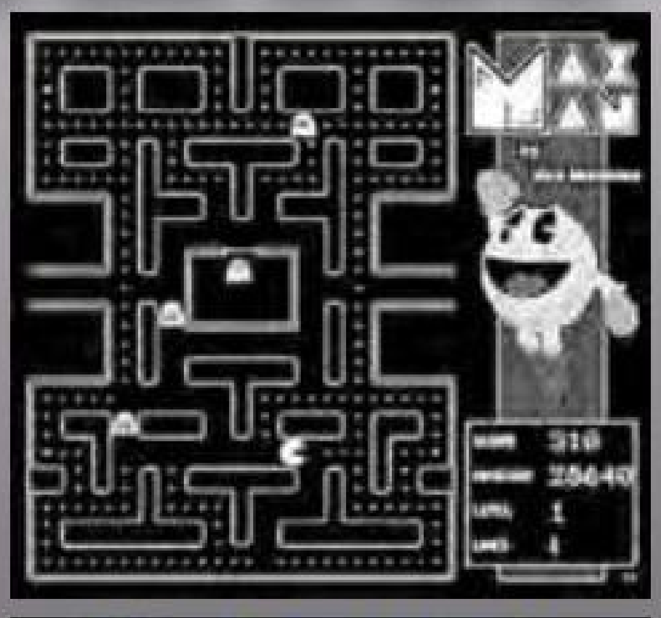
Monday to Friday 08:30 - 17.00, Saturday 08:30 - 15:30

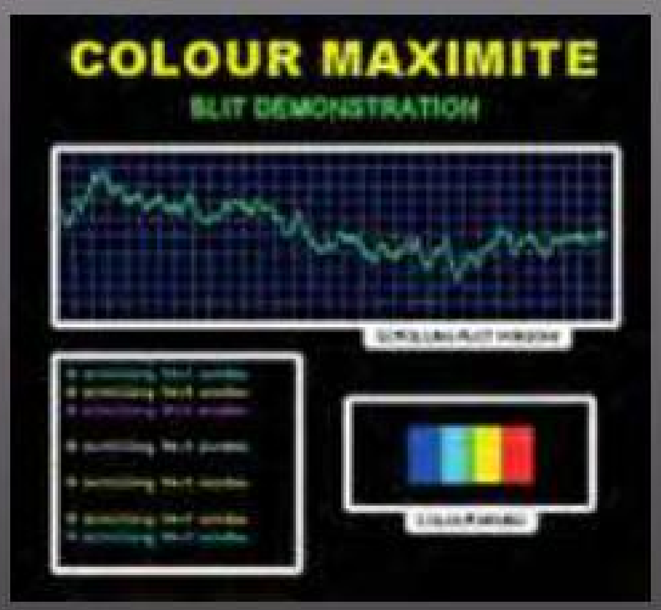
Station Road
Cullercoats
North Shields
Tyne & Wear
NE30 4PQ

Tel: 0191 2514363 sales@esr.co.uk www.esr.co.uk

Colour Maximite Computer

COLOUR MAXIMITE
MAXIMITE
SD





Part 1

Words: Phil Boyce
Design: Geoff Graham

A retro 80s home computer with modern-day features

Build your own standalone computer based on a powerful PIC32 microcontroller running the easy-to-use MMBASIC language.

The Colour Maximite is a small, self-contained computer that is ideal for experimenting, learning to code and for using as an embedded controller. It uses a standard VGA monitor for a display, a PS2 keyboard for input and an SD card for storage. It is programmed in BASIC, which makes the *Colour Maximite Computer* extremely easy to use. This DIY project only takes a couple of hours to build and will provide endless fun – it can even play some classic arcade games. Before we explore its capabilities, let's first rewind the clock 35+ years (and for our older readers, have a trip down memory lane).

Computers enter the home

If you were lucky enough to own one of the first home computers back in the early 1980s, then you will probably remember having to plug it into the aerial socket of your bulky TV (or into a CRT monitor if you were fortunate enough to have one), connect up the power pack, and then switch it on. Once you had correctly 'tuned in' your TV (which often meant flexing your body over to the rear of the TV – 'dust magnet'), you were then presented with a 'Teletext-style image', probably stating the model of your computer and also letting you know how much memory it had available. The image on the TV would have been rather blocky, and if you were fortunate to have a colour computer and a colour TV, then it would have looked absolutely amazing! A (flashing) cursor signified that your computer was waiting for you to do something – and at this point you could immediately start typing.

These home computers really were great fun to use. They enabled you to explore how to code (in BASIC, and in some cases, machine code), and at a moments notice, you could choose to load one of the many games that flooded the market. Sure, each of these computers (Sinclair, BBC, Commodore, Amstrad, Atari, to name but a few) had their individual quirks, such as slow (and often unreliable) loading of a program (from cassette, disk, or micro-drive), many had rather poor image quality and/or resolution, and some had sound capabilities that stretched to nothing more than a simple beep. You may have purchased your home computer because you wanted to connect it to your own external hardware and control it via a program. This required an understanding of how to 'access' the I/O connector, often located on the back of the computer. Typically, this was doable (and often using the `POKE` command), but it was not always a simple task, and certainly not without risk of permanently damaging your computer.

The important point is that these home computers were great fun to use; and they helped create a generation of users, many of whom went on to have successful careers in IT. As an aside, let's not forget several popular TV series broadcast on the BBC that taught people how to use their 'shiny new computer'. My favourite was *The Great Egg Race*, which involved competitors using their home-built robot-buggies to navigate through a physical maze.

General and system users

Returning to the present, computer users today typically fall into one of two categories:

1. General users – running 'apps' to achieve an end result
2. System users – developing a solution; typically involving a dedicated development environment running on a PC, Mac or Linux machine.

Anyone browsing the internet, or using a wordprocessor, spreadsheet, database, dedicated end-solution, social media, watching a movie or playing the latest video game will fall into the first category of 'general user'.

People developing any of the above applications will fall into the second category, including web developers, and also users of single-board computers (SBCs) such as the very popular Raspberry Pi, Arduino, and MicroBit. Those in this second category are typically (but not always) 'systems people', and at sometime will have had a learning curve to overcome in order to be able to 'create' – not something a total novice can immediately do. It is worth mentioning here that the 'maker and hacker' environment has increased the popularity of microcontrollers and SBCs – although strictly speaking these users should be classified as 'geeky system users' rather than 'general users'.

The need for skills

Since the 1980s, and as technology has moved on, the skill requirement between a 'general user' and a 'system user' has widened. In response to this, several years ago the UK government said it needed 'youngsters' to learn the skills needed to write apps (ie, learn to code) rather than just being able to use apps. This led to computer science being introduced into school curricula. The idea was also to promote 'creativity' (and digital design) – vital skills for boosting any modern economy.

Computer science was a great idea, but unfortunately it was not well implemented. Even today, most schools are still not teaching creativity, let alone teaching coding. The main excuse often expressed is a lack of simple-to-use hardware (for teachers, as well as pupils); and also a lack of suitably trained staff.

Fortunately, it is not all bad; there are many fantastic IT opportunities outside school. However, you do have to go and look for them, and they are often aimed more at IT professionals rather than school children, which is the group the government wants to target. Bearing in mind recent advances in technology, wouldn't it be fantastic to have another 'boost' similar to that back in the 80s, so that a new generation could grow up being creative, inspired, and exited by designing innovative technological solutions?

A product that would allow anyone to quickly and simply convert an idea into a prototype really could excite a new generation of creative entrepreneurs. It would make sense to base such a product on all the benefits of an 80s home computer, but which also improves in areas that weren't quite so great. It would need to be a product that is easy to use, with a minimal learning curve – even for beginners. It should be a standalone machine, instantly ready to use when it is switched on, and have the ability to save and load programs quickly and reliably. It also needs to have a wide range of connectivity options to the outside world (allowing for easy interfacing to the vast array of hardware modules that are available today). And, ideally it needs to be low cost. Finally, why not make it available in kit form, so that anyone interested in building it can do so (just as the Sinclair ZX81 was offered as a kit in 1981 for £49.95 – nearly £200 in today's money).

Table 1: Features and specifications

- Self-contained, high-performance, DIY, standalone computer
- Instantly on – ready to use the moment power is applied
- Uses a single, powerful 100-pin PIC32 microcontroller running at 80MHz
- Fully featured BASIC interpreter and programming language (MMBASIC)
- 512K Flash memory and 128KB RAM

- VGA socket providing eight colour VGA output (black, red, green, blue, cyan, magenta, yellow, white)
- Four graphics modes:

Mode 1: fast	480 × 432 pixels, two-colour mode
Mode 2: medium-speed	480 × 432 pixels, four-colour mode
Mode 3: slowest	480 × 432 pixels, eight-colour mode
Mode 4: high-speed	240 × 216 pixels, eight-colour 'games mode'

- Graphics modes can be changed within program code, allowing full flexibility of speed, memory use and colour quantity
- Graphical features include: selectable fonts, user-designed fonts and commands PIXEL, LINE, CIRCLE, BLIT (high-speed copying of a selectable part of the screen) and SPRITE (for creating animation without disturbing the background)

- Audio output socket (3.5mm)
- Stereo sound output with TONE command (generates accurate sinewave)
- High quality stereo music synthesiser with PLAYMOD command (to play music / sound-effect MOD files)
- Alternatively, use as a 2-channel PWM-driven analogue voltage output

- 40 available I/Os covering: digital inputs, digital outputs, analogue inputs, frequency and voltage measurements, serial/UART, SPI, I^2C, 1-wire
- 20 available on rear panel – accessed via a 2 ×13 connector (includes 4 × 0V output, 3.3V output, and 5V output)
- 20 presented on an internal Arduino footprint connector – can plug in Arduino shields/protoboards

- Full-size SD socket for program/data storage (up to 32GB)
- SD file system uses FAT16 and FAT32 file system – useful for editing programs on a 'bigger computer' and transferring Maximite data into a spreadsheet/database
- PS2 socket for standard PS2 keyboard input
- Battery-backed real-time clock (RTC)

- USB socket (Type B) for connecting 5V power. Can also be used to connect to a PC/MAC/Linux (used as an alternative method to control/program the Colour Maximite Computer via a VT100 compatible application such as TeraTerm or PuTTY)
- Powered via USB (5V), or via barrel-jack connector (7-16V; 9V max recommended)

- Attractive ABS enclosure, measuring 140mm × 35mm × 110mm (w × h × d)
- Pre-cut front and rear panels

Colour Maximite Computer

Enter the *Colour Maximite Computer* – a single-chip, low component-count design that meets all the above criteria. By the way, users of the powerful Micromite controller (and those following the *Make it with Micromite* (*MIWM*) series) will absolutely love the *Colour Maximite Computer*, as it uses its own version of MMBASIC – a programming language that is powerful, yet extremely easy to master.

The *Colour Maximite Computer* is yet another exceptional design from the very talented Geoff Graham (**geoffg.net/maximite.html**). Simply plug in a standard VGA monitor (instead of an old CRT TV), insert a PS2 keyboard, connect power (from a USB power source) and you have a small, self-contained computer with absolutely no need to tether it to a larger computer. Put simply, it is a retro 80s home computer incorporating modern features, most notably the ability to simply add and control external hardware (should you need to). Switch it on and it is immediately ready to use. Programs can be stored on an SD card (much more reliable than cassettes, disks and microdrives), and if you connect an external amplifier and speakers then you will hear its amazing sound capabilities. So, whether you want to learn how to code in BASIC, write a custom program, experiment with electronics, play a classic game, or control external hardware, the *Colour Maximite Computer* will help you achieve this (and a lot more besides).

This month, we will discuss some more of the specification and features of the *Colour Maximite Computer*, including a brief description of how it works.

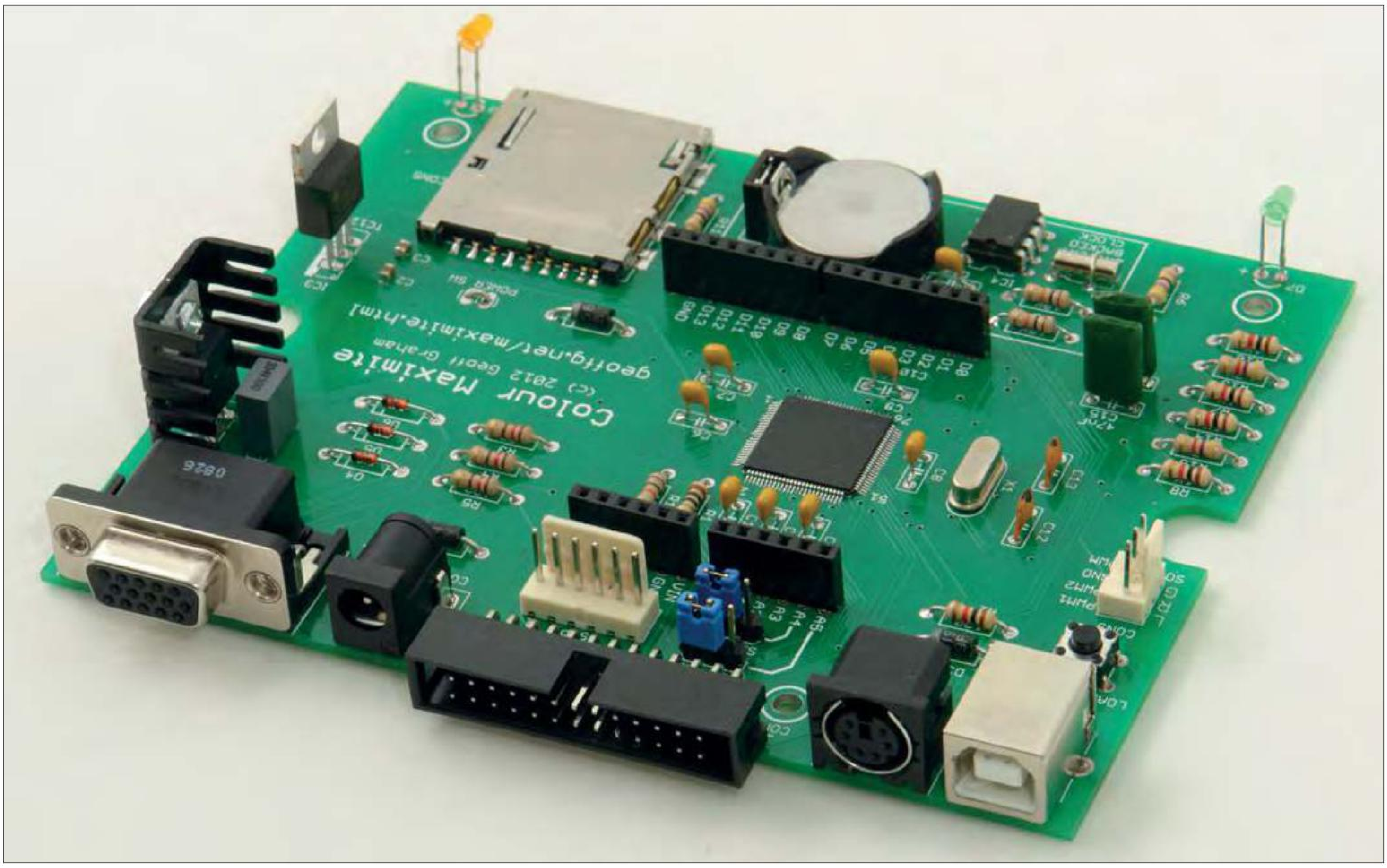


Fig.1. The *Colour Maximite Computer* is based around a 100-pin PIC32 microcontroller. This is an SMD, as is the SD socket and 10µF capacitor on pin 85; all other components are through-hole. (The above is a prototype which varies slightly from the final design)

Next month, we will take you through the details of the construction (for those who want to assemble it themselves), and the month after we will give you some ideas for how to use the *Colour Maximite Computer* (including how to have some fun playing classic games). I hope you are as excited as we are about what is coming up.

Before we begin, I need to highlight what the *Colour Maximite Computer* is *not*. Some 'professionally trained' systems people will insist that you must only use big-name branded development tools (and programming languages) to achieve anything worthy, and also to only design a solution that connects to the Internet (or cloud) so it can be accessed from anywhere in the world. Well, this is not our belief, and it is not what the *Colour Maximite Computer* is about in any way, shape, or form. There is no Internet connectivity, so it cannot be used to browse the web, and there are definitely no big-name software tools that need to be purchased, installed, or used. So, if you're looking for a cloud-based solution, programmed in the latest must-use programming language using a new (and potentially costly) development tool, then you will be thoroughly disappointed. However, if you want to have lots of fun exploring and creating new ideas, or even building a project that you've been thinking about for some time, then please do read on – you won't regret it.

Features and specifications

Table 1 provides a summary of the features and specifications of the *Colour Maximite Computer*.

The amazing thing about this compact computer is that everything (VGA timing, VGA colour signals, keyboard input, USB interface, sound output, music playback, I/O control, SD card interface, interpreter, editor…) is controlled by the firmware running on a single chip: a PIC32MX microcontroller. Furthermore, there are only three surface-mount devices (SMDs) used in the *Colour Maximite Computer* – the PIC32 microcontroller (a 100pin 14mm × 14mm TQFP package), a full-size SD socket, and a single 10µF ceramic capacitor (dedicated to stabilising the PIC32s internal 1.8V supply). Only a small number of additional support components are required (see the circuit diagram in Fig.5), and these are all through-hole components, making construction of this project extremely easy (as we will see next month).

MMBASIC

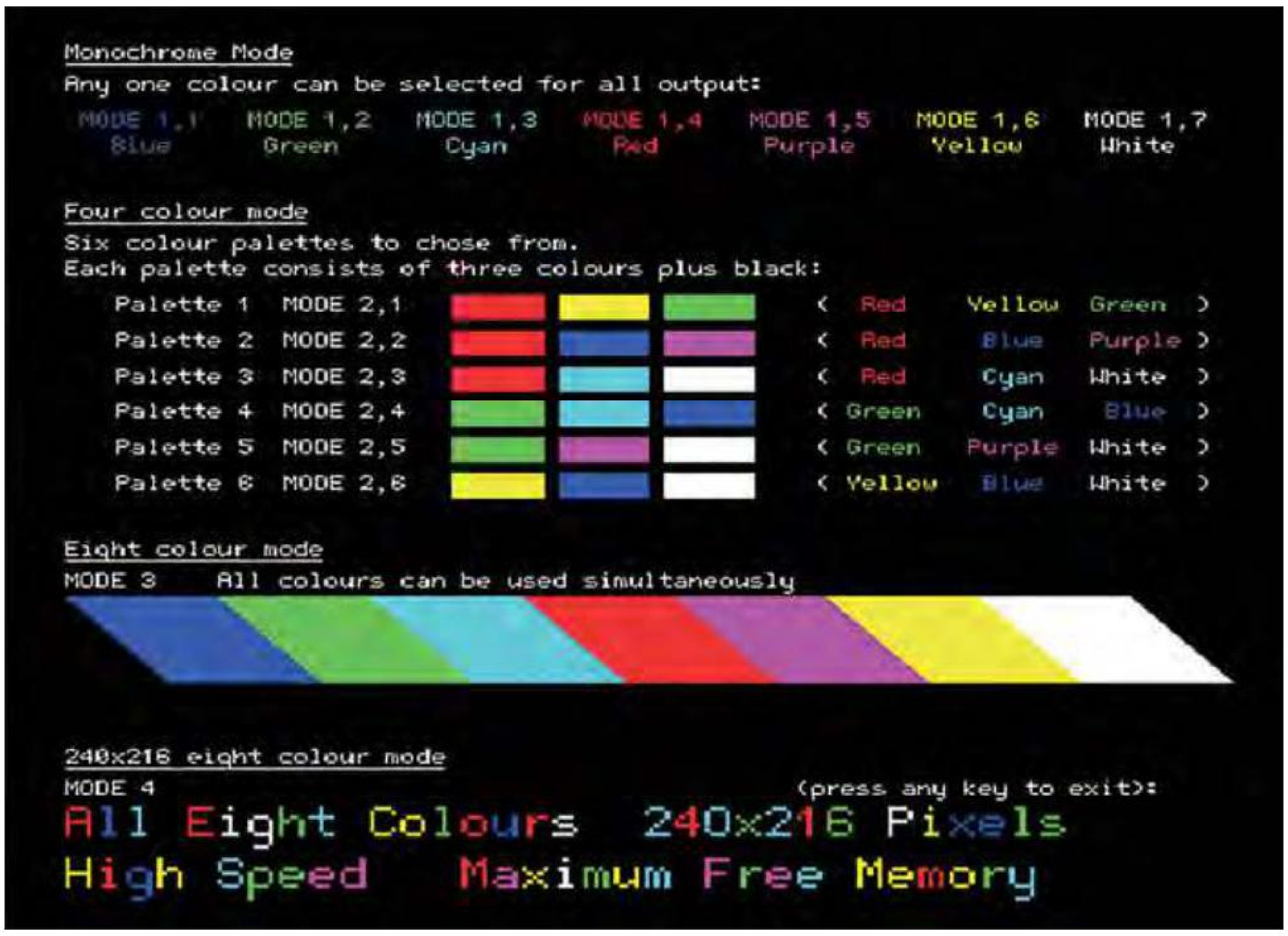
The *Colour Maximite Computer* is programmed in MMBASIC (which is based on Microsoft BASIC). BASIC is an easy-to-learn intuitive language, and readers following the *MIWM* series will find the *Colour Maximite Computer* version of MMBASIC very similar to that used on the Micromite.

MMBASIC features a full-screen editor, allowing you to directly enter (and edit) your program code. To test your program, simply type `RUN`, and to stop it press `Ctrl-C`. Typing `EDIT` will launch the full-screen editor, allowing you to make further changes to your program code. Programs can be `SAVEd` onto the SD card; likewise a program can be `LOADed` from the SD card. Note that at any one time, only a single program can be in the *Colour Maximite Computer's* memory (RAM).

Because MMBASIC is very easy to learn, it is an ideal introduction to programming. It also means that users wanting to 'play' with external hardware can easily do so without the steep learning curve that other programming languages typically require. But don't let this lead you into a false impression about MMBASIC – it is a very powerful language when you need it to be (once you realise all the commands and features that are at your disposal).

MMBASIC supports floating-point numbers, long variable (and string) names, multi-dimensional variable (and string) arrays, subroutines, functions, interrupts, as well as commands specifically aimed at controlling hardware, including

`IR`	Infrared send and receive
`LCD`	Interfacing character LCDs
`KEYPAD`	Interfacing matrix keypads
`DISTANCE`	Interfacing with ultrasonic range modules



```
Monochrome Mode
Any one colour can be selected for all output:
 MODE 1,1   MODE 1,2   MODE 1,3   MODE 1,4   MODE 1,5   MODE 1,6   MODE 1,7
   Blue      Green       Cyan        Red      Purple     Yellow      White

Four colour mode
Six colour palettes to chose from.
Each palette consists of three colours plus black:
   Palette 1  MODE 2,1   🟥 🟨 🟩   <  Red     Yellow  Green  >
   Palette 2  MODE 2,2   🟥 🟦 🟪   <  Red     Blue    Purple >
   Palette 3  MODE 2,3   🟥 🟦 ⬜   <  Red     Cyan    White  >
   Palette 4  MODE 2,4   🟩 🟦 🟦   < Green    Cyan     Blue  >
   Palette 5  MODE 2,5   🟩 🟪 ⬜   < Green   Purple   White  >
   Palette 6  MODE 2,6   🟨 🟦 ⬜   < Yellow   Blue    White  >

Eight colour mode
MODE 3    All colours can be used simultaneously


240x216 eight colour mode
MODE 4                                   (press any key to exit):
All Eight Colours   240x216 Pixels
High Speed   Maximum Free Memory
```

Fig.2. Screenshot showing the eight available COLOURs and the four screen MODEs.

`DS18B20`	Interfacing temperature sensors.

For much more detail, download the *Maximite Language Manual* from: **http://bit.ly/pe-nov19-max**

Graphics

There are up to eight COLOUR choices available on the *Colour Maximite Computer*, along with four different graphics `MODEs` (with a maximum pixel resolution of 480 × 432) – see Table 1. MMBASIC graphical commands allow you to control the colour of an individual pixel, and to draw lines, boxes and circles. `BLIT` allows you to quickly copy one part of a screen directly to another. `SPRITEs` are also possible, which are extremely useful when writing games where you need to move a character (ie, 'sprite') around the screen. A `SPRITE` is a 16 × 16 block of pixels that can be drawn on the screen without affecting the 'background'.

Fig.2 shows the eight colours and four graphics modes in action on a single screenshot. Everything is coded into the Firmware on the PIC32, so there is a small trade-off between speed, memory use, resolution and colour quantity. That said, some very impressive results can be obtained with these resources.

Simplicity: a *Colour Maximite Computer* set-up

So how does the PIC32 (with MMBASIC) create the screen image? The following description is simplified, but should shed enough light on how it all works. A VGA colour monitor is driven by signals on three colour inputs: red, green and blue (hence 'RGB'); along with two more signals that receive horizontal and vertical sync pulses, which when sent at specific timings result in a 'stable' display. By driving the three colour signals either high (fully on), or low (fully off), you can obtain various colours on the screen.

Driving any one-colour signal high results in that colour being displayed (so driving just the red signal high will result in red pixels being displayed). Driving any two results in a 'mix' of those two colours. So mixing red and blue gives magenta, mixing red and green give yellow, and mixing blue and green gives cyan. And if you were to mix all three-colour signals, then the result is white (see Fig.3).

Finally, with no signals set high the result is 'black'. Hence these are the eight colours that are available on the *Colour Maximite Computer*. For the technically minded, the three colour signals are driven high or low by three separate SPI outputs on the PIC32, with the 'pixel data' (video bits) being clocked out from video memory at 20MHz onto each colour signal. So from the video buffer and DMA controller, this equates to 60 million bits per second – not bad, bearing in mind this all happens effectively in the background! By the way, the horizontal and vertical sync pulses are generated by internal interrupts (way beyond the scope of this article) and ultimately dictate the overall pixel resolution.

Stereo audio, and analogue voltage output

A 3-way PCB header connects to a 3.5mm stereo audio output socket mounted on the rear panel. Typically, this socket is used to connect an external amplifier and speakers. Using the `TONE` command, you can generate highly accurate sinewaves at audible frequencies, and hence you can easily write a program to play musical tunes. You could also use the `TONE` command to create DTMF signals. However, for some really impressive sounds, you want to use the `PLAYMOD` command. This starts an internal music synthesiser, which is able to read `MOD` sound files – a file format that was commonly used with the brilliant sounding Atari ST. With the capability of `MOD` playback, you are able to incorporate fantastic sound effects into your programs, as well as liven up games by playing background music.

As an alternative to 'stereo audio' output on the 3.5mm connector, the `PWM` command can be used instead to drive external hardware with two analogue voltage signals between 0V and 3.3V – all under program control. Note: if you intend to use the stereo output connector with the `PWM` command, do *not* use any of the audio commands! Depending on exact usage requirements, there may be a need to change the values of some of the passive components in the default audio circuit (to create a suitable low-pass filter to average out the digitally controlled duty cycle of the PWM signal). More details will be explained about this next month when we discuss the construction.

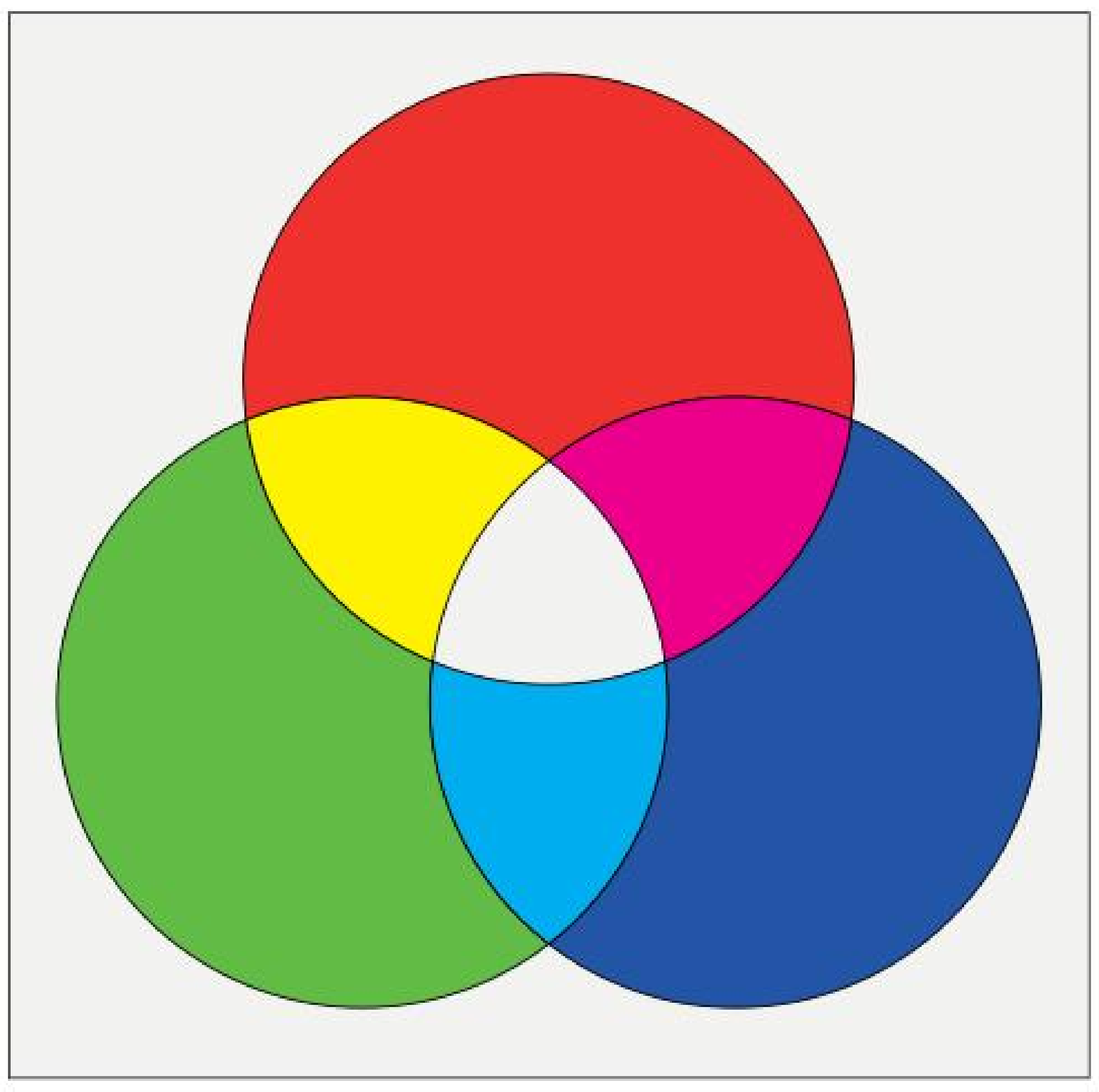


Fig.3. Combinations of the red, green, and blue signals produce the seven colours. Black makes the eighth colour when all three RGB signals are off.

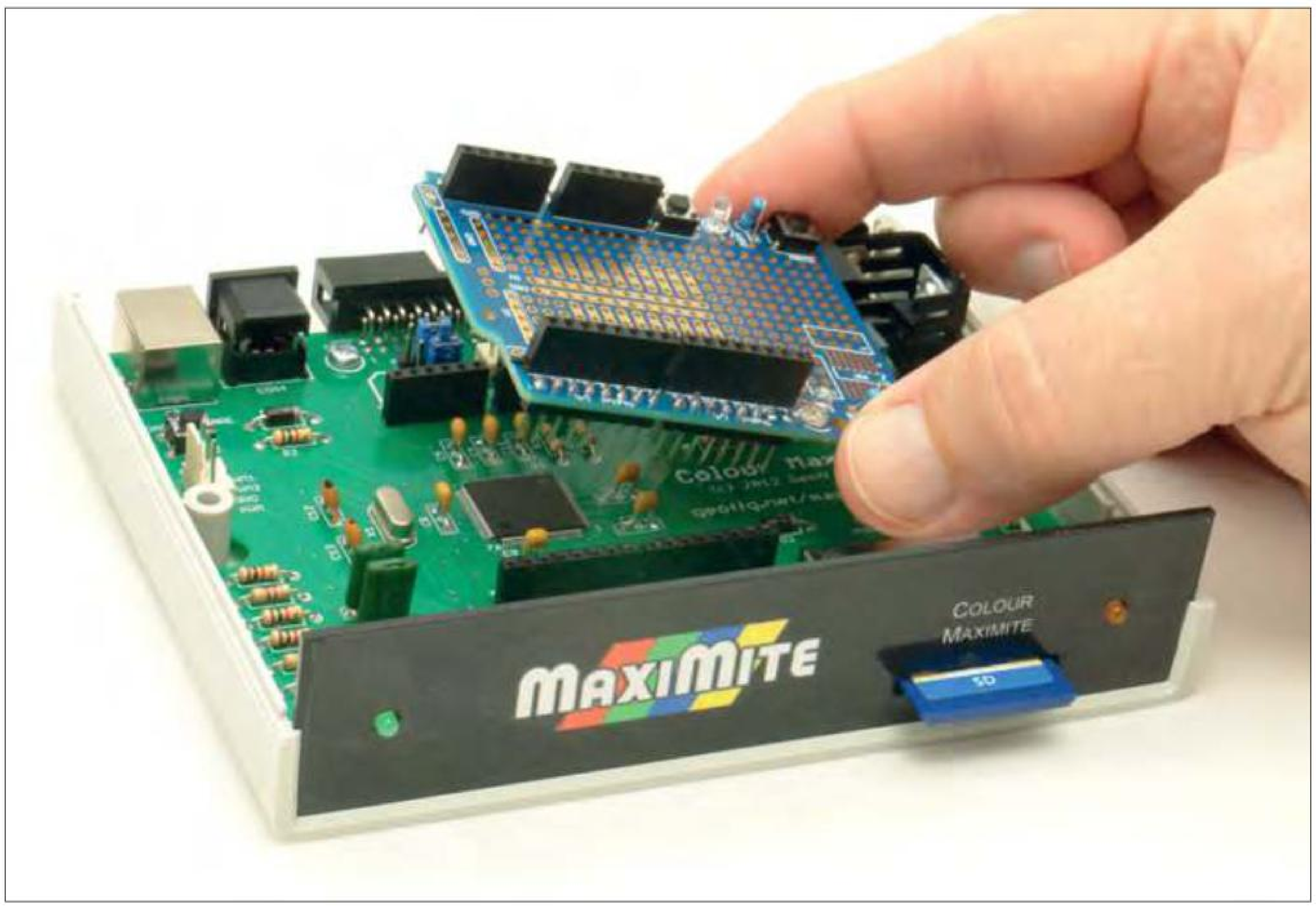


Fig.4. The *Colour Maximite Computer* is designed to accept an Arduino shield via an internal I/O connector. Shown here, a prototyping shield.

I/O capabilities

The various protocols that are compatible with MMBASIC are listed in Table 1. Hardware that uses any of these can be interfaced with the *Colour Maximite Computer*. There are 40 inputs and outputs, split across two connectors – one external, and one internal.

The external connector is available on the rear panel and is typically used with a 26-way ribbon cable running directly to your own hardware. The internal connector has the same footprint as an Arduino and allows you to directly plug in Arduino Shields (see Fig.4). It is worth noting that the main idea behind the internal Arduino footprint connector is to provide a simple method to add your own hardware permanently inside the *Colour Maximite Computer's* case by use of a prototyping shield. When it comes to 'shields', there are three types to consider:

1. Prototyping – unpopulated boards
2. Simple I/O – relays, buttons, buzzers, LEDs…
3. 'Complex' – eg, MP3 players.

The first two types of shield are extremely easy to use with the *Colour Maximite Computer* since they do not require a software driver to operate; however, 'complex' shields will probably need a driver of some sorts to function correctly. It may well be possible to use a 'complex' shield, but you would need to write your own driver in BASIC (and hence it would not operate as fast as a driver written in the C language) – I'm highlighting this just in case you wish to use a complex shield, which normally comes with its own custom Arduino software driver.

Battery-backed clock

There is provision on the *Colour Maximite Computer's* PCB for a DS1307 real-time clock (RTC) chip, along with its associated components, a 32.768kHz crystal and CR2032 lithium coin-cell battery. The RTC runs independently of the PIC32 and will keep track of the time and date when you remove power from the computer. Note that there is a planned modification to incorporate the superior DS3231 RTC module because these suffer from less temperature drift – or put another way, they are far more accurate in keeping the correct time when compared to the DS1307.

The RTC chip is optional and is not required if using the *Colour Maximite Computer* for tasks such as learning to code or playing games. However, for data logging-type use, we recommend that you install the RTC chip (and associated components). When the *Colour Maximite Computer* is powered up, it first checks to see if the RTC chip is installed. If it is, then it reads the time and date from the battery-backed RTC and loads this information into the PIC32's own crystal-locked clock. From then on, whenever a program needs to know the time and/or date, it is retrieved from the PIC32's internal registers (which are faster in operation than retrieving the information over the relatively slow I^2C bus used by the RTC). If the RTC option is not installed, then `TIME$` and `DATE$` can be set manually; however, this information will be lost as soon as you remove power from the computer.

Power supply

The *Colour Maximite Computer* typically draws around 160mA at 5V (when not driving any external hardware). There are two available options to power the *Colour Maximite Computer*. The recommended option is to simply use a 5V USB PSU. A USB phone or tablet charger is perfect for this task, and these are readily available – however, do check that the output voltage is indeed 5V (some PSUs have a slightly higher output voltage to compensate for voltage drop in thin USB cables).

The much-less-preferred option for powering the *Colour Maximite Computer* is to use a 7-16V DC PSU connected to the rear-mounted barrel-jack connector. So why do we not recommend this method? It comes down to 'wasted energy'. Internally, the *Colour Maximite Computer* contains two voltage regulators: a 5V regulator (with attached heatsink), which in turn has its 5V output feeding into a 3.3V regulator. The 5V regulator only comes into play when external power is applied via the barrel-jack. In this scenario, the 7-16V PSU feeds into the 5V voltage regulator, and even though the regulator is specified up to 16V input (and it has an attached heatsink), there is the potential for a great deal of internal heat build up. So if you have to use a DC power supply via the barrel-jack connector, it is strongly recommended that you keep the input voltage below 9V.

However it is derived, the internal 5V is used to power the attached PS2 keyboard, supply and the RTC chip (if installed), and it is also presented to both the external 26-way I/O connector, as well as the internal Arduino connector (to power any additional 5V hardware). As mentioned above, the 5V also feeds into the 3.3V regulator, which in turn powers the PIC32 and the SD card (when inserted). The 3.3V supply is also presented to the I/O connectors for powering additional 3.3V hardware.

A third regulator is also present; however, this is inside the PIC32 microcontroller and it simply supplies the required internal 1.8V for the microcontroller to operate correctly. The generated 1.8V is available on Pin 85 (Vcap) of the PIC32, and requires a 10µF ceramic capacitor to be connected between this pin and 0V to suppress noise and stabilise the internal regulator. Note that this 10µF ceramic capacitor is essential for the PIC32 to operate correctly and it must not be left out. It is mentioned here because this is a small SMD component and can easily be overlooked.

One final item worth mentioning while discussing the power supply is that there are several 100nF capacitors on the PCB located around the PIC32. They are each connected across the

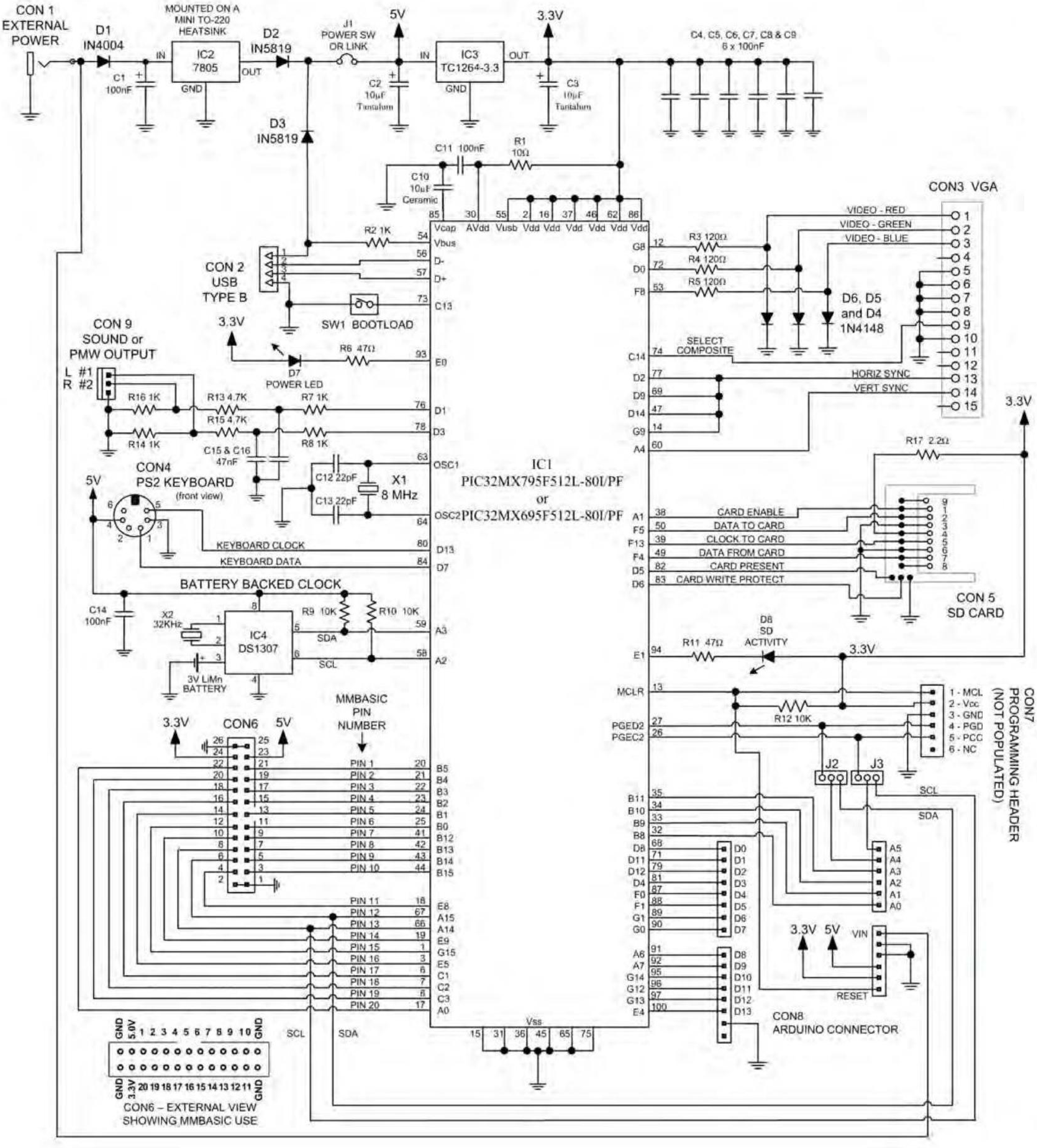

CON 1 EXTERNAL POWER
D1 1N4004
MOUNTED ON A MINI TO-220 HEATSINK
IC2 7805
IN
OUT
GND
C1 100nF
D2 IN5819
D3 IN5819
J1 POWER SW OR LINK
5V
C2 10µF Tantalum
IC3 TC1264-3.3
3.3V
C3 10µF Tantalum
C4, C5, C6, C7, C8 & C9 6 x 100nF
C11 100nF
R1 10Ω
C10 10µF Ceramic
CON3 VGA
VIDEO - RED
VIDEO - GREEN
VIDEO - BLUE
R3 120Ω
R4 120Ω
R5 120Ω
D6, D5 and D4 1N4148
SELECT COMPOSITE
HORIZ SYNC
VERT SYNC
CON 2 USB TYPE B
R2 1K
SW1 BOOTLOAD
CON 9 SOUND or PMW OUTPUT
L #1
R #2
R6 47Ω
D7 POWER LED
R16 1K
R13 4.7K
R7 1K
R15 4.7K
R14 1K
R8 1K
C15 & C16 47nF
C12 22pF
C13 22pF
X1 8 MHz
IC1 PIC32MX795F512L-80I/PF or PIC32MX695F512L-80I/PF
CON4 PS2 KEYBOARD (front view)
KEYBOARD CLOCK
KEYBOARD DATA
BATTERY BACKED CLOCK
C14 100nF
X2 32KHz
IC4 DS1307
3V LiMn BATTERY
R9 10K
R10 10K
SDA
SCL
R17 2.2Ω
CARD ENABLE
DATA TO CARD
CLOCK TO CARD
DATA FROM CARD
CARD PRESENT
CARD WRITE PROTECT
CON 5 SD CARD
R11 47Ω
D8 SD ACTIVITY
R12 10K
CON7 PROGRAMMING HEADER (NOT POPULATED)
1 - MCL
2 - Vcc
3 - GND
4 - PGD
5 - PCC
6 - NC
J2
J3
MMBASIC PIN NUMBER
CON6
PIN 1 … PIN 20
CON8 ARDUINO CONNECTOR
VIN
RESET
A5 A4 A3 A2 A1 A0
D0–D13
Vss
CON6 – EXTERNAL VIEW SHOWING MMBASIC USE
GND 5.0V 1 2 3 4 5 6 7 8 9 10 GND
GND 3.3V 20 19 18 17 16 15 14 13 12 11 GND
SCL
SDA

Fig.5. Full circuit diagram of the *Colour Maximite Computer.*

3.3V supply rail and are used to suppress noise generated by the PIC32 while it is busy working through the many tasks it needs to perform.

Next month

In *Part 2*, we will provide full construction details of the *Colour Maximite Computer* kit (available from **micromite.org**). The end result will be a fully functioning computer that will open up a world of possibilities. In the meantime, do take a look at this video review of the Colour Maximite by '8-Bit Guy' (**https://youtu.be/XQA8lowEKOo**). For those of you unfamiliar with him, he reviews a lot of 'retro' technology and has a massive Internet following. When he initially posted this video, he got around 100,000 views in the first week alone!

MAXIMITE

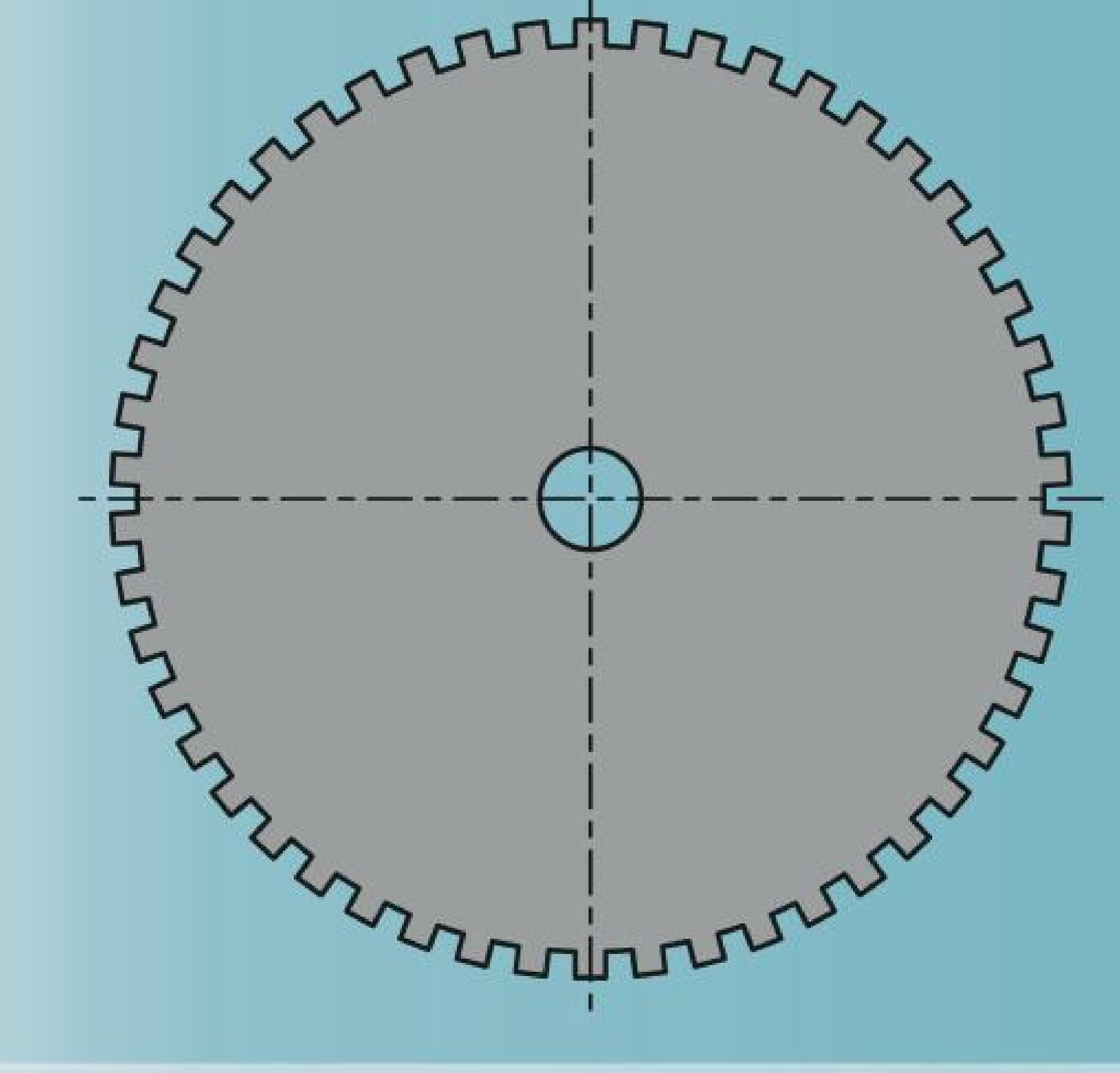
Colour Maximite

Schematic V1

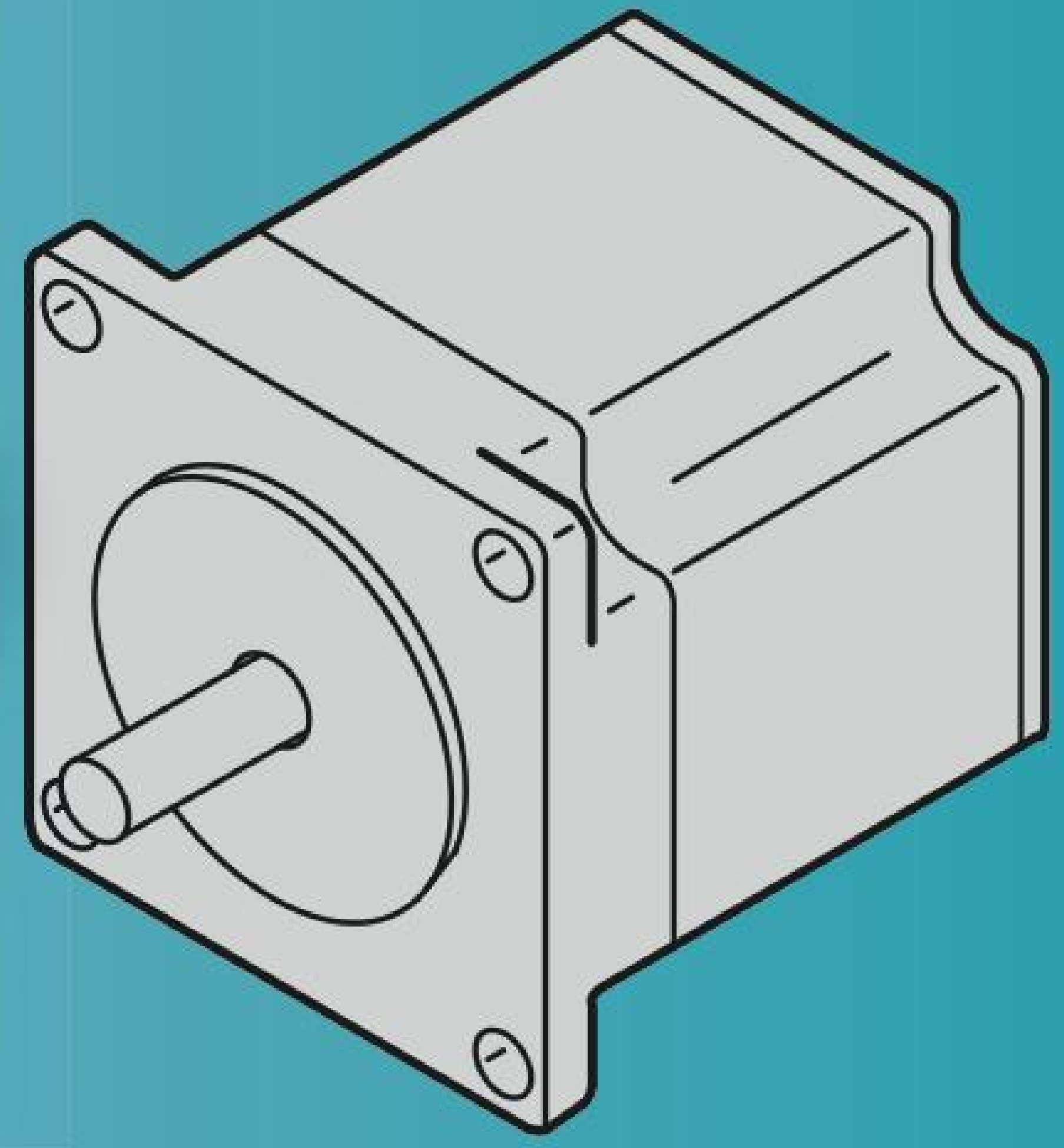
Geoff Graham

http://geoffg.net/maximite.html

This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Australia (CC BY-NC-SA 3.0)



Using Stepper Motors



Part 2 | by Paul Cooper | technobotsonline.com

Choosing and identifying stepper motors

This month, we will look at choosing a stepper motor and provide an in-depth guide on how to identify the leads on 4, 5, 6 or 8-lead hybrid motors.

Choosing a stepper motor

There a number of parameters you need to decide upon before choosing the right stepper motor for your project. Unfortunately, some of these parameters can be difficult to accurately predict or calculate without getting really bogged down with mathematics and computer modelling. Specifying a conventional DC brushed motor or geared motor would usually come down to voltage, RPM and torque. Torque for a DC motor would be specified at a particular RPM and may even include a stalled motor torque figure. Stalling a DC brushed motor can quickly lead to the windings over-heating, smoke and failure.

Stepper motors are very different in that they are designed to operate from zero to some maximum RPM. Maximum torque is at zero RPM and is referred to as the holding torque. As the RPM is increased, the available motor torque drops off. Try to step a stepper motor too fast to the point where it physically cannot keep up, then it will miss steps or 'stall'. This can also occur by trying to accelerate a stepper motor too quickly.

So, returning to what parameters we as hobbyists may need to consider:

- Holding torque
- Required maximum RPM
- Pull-out torque, the load a motor can move at the required maximum RPM
- Step size in degrees or steps per revolution.

In addition to the above, the rate of motor acceleration may be important, in which case the motor inductance is also a consideration, but more on this later. Once you have found a motor that meets your needs, it will quote the phase current at a particular voltage which is used to size the drive electronics.

A definitive guide on identifying the wires on a stepper motor

Don't have the datasheet to hand or have a surplus or salvaged motor with no manufacturer's model number marked on it? No problem – with just a multimeter, you can identify the leads on a 4-lead and 6-lead motor before connecting to the driver. 5-lead is not so easy and for an 8-lead you will need a dual channel oscilloscope!

4-lead stepper motors

This should mean it's a 2-phase bipolar design, as per Fig.7a. It's unlikely, but it could also be a variable reluctance motor; but we'll check that during these tests.

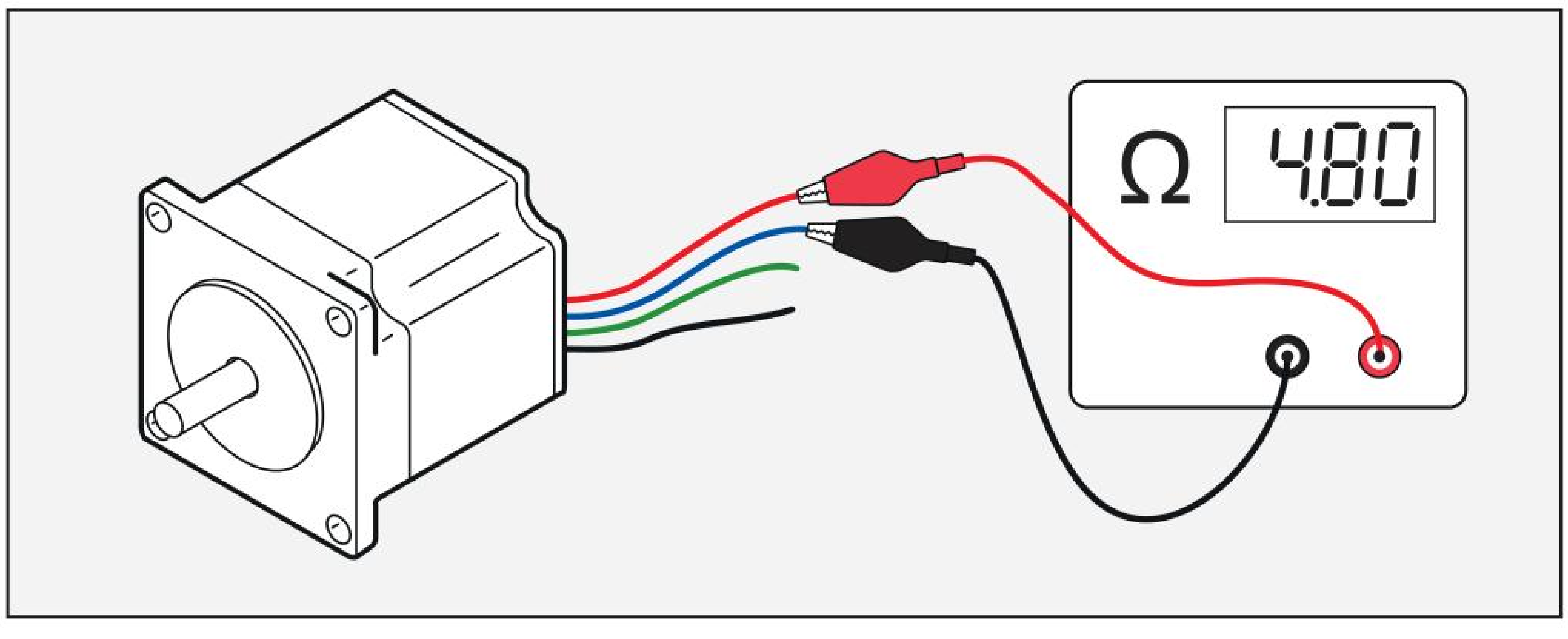


Fig.11. For a 4-lead motor, first find a pair that gives a low resistance.

Using a multimeter on its 200Ω resistance range, clip the red test lead on any of the four leads. Then using the other test lead, check the resistance on each of the three other leads (Fig.11).

One wire should give a low resistance, the other two wires no reading / open circuit. The leads giving the low resistance are one phase, the remaining two leads are the other phase, but double check their resistance as well as they should have an identical resistance to the first pair. Drivers may well be marked 1A 1B (for phase 1) and 2A 2B (for phase 2) or similarly A+ A–, B+ B–, as long as you wire the phases in the correct pairs, the polarity of a phase or whether it is phase 1 or phase 2 does not matter. All that will happen is the direction of rotation could be wrong, which is easily rectified by swapping around *two* of the leads on only *one* of the phases.

Lead colour	Lead colour	Resistance (Ω)
Pink	Orange	60
Pink	Yellow	60
Pink	Red	30
Pink	Blue	60
Orange	Yellow	60
Orange	Red	30
Orange	Blue	60
Yellow	Red	30
Yellow	Blue	60
Red	Blue	30

Table 2: In this example, the four half-value resistance readings have the red lead in common, so the red lead is the centre tap.

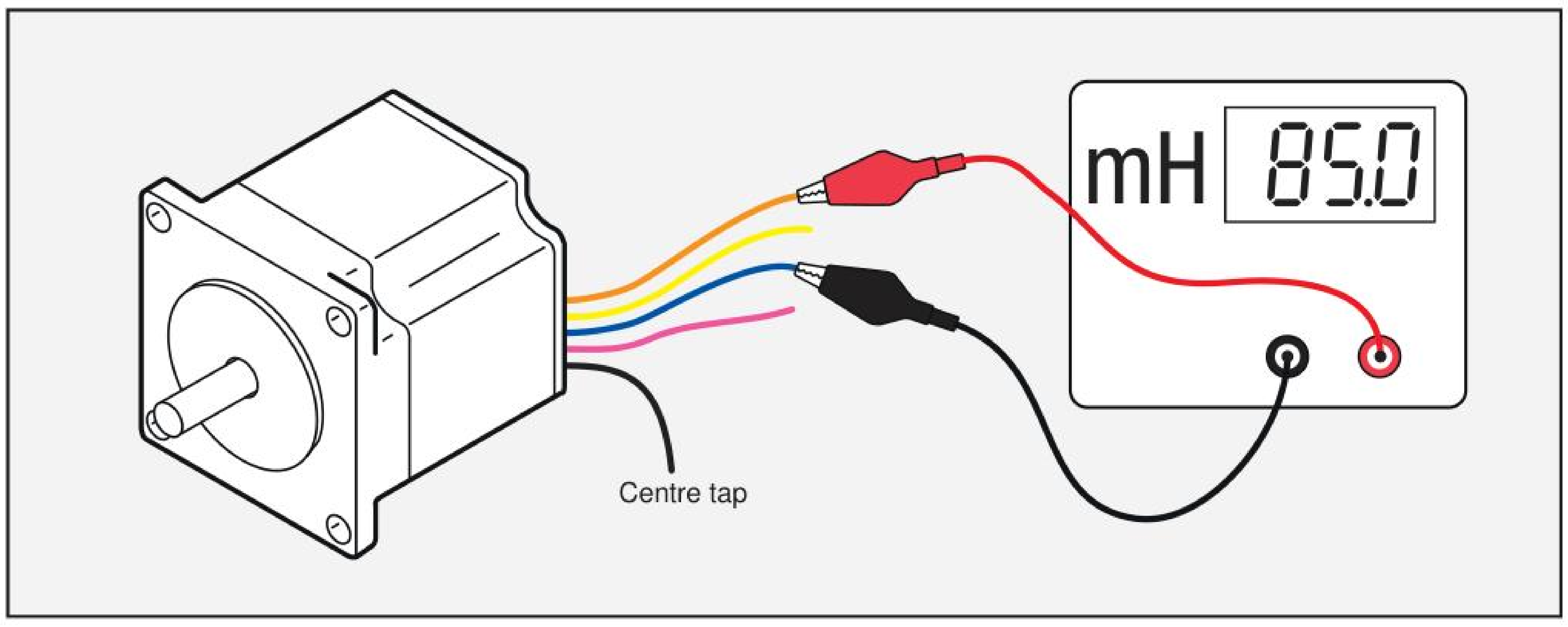


Fig.12. Measuring the inductance of the 4 wires to identify those on the same common magnetic flux path.

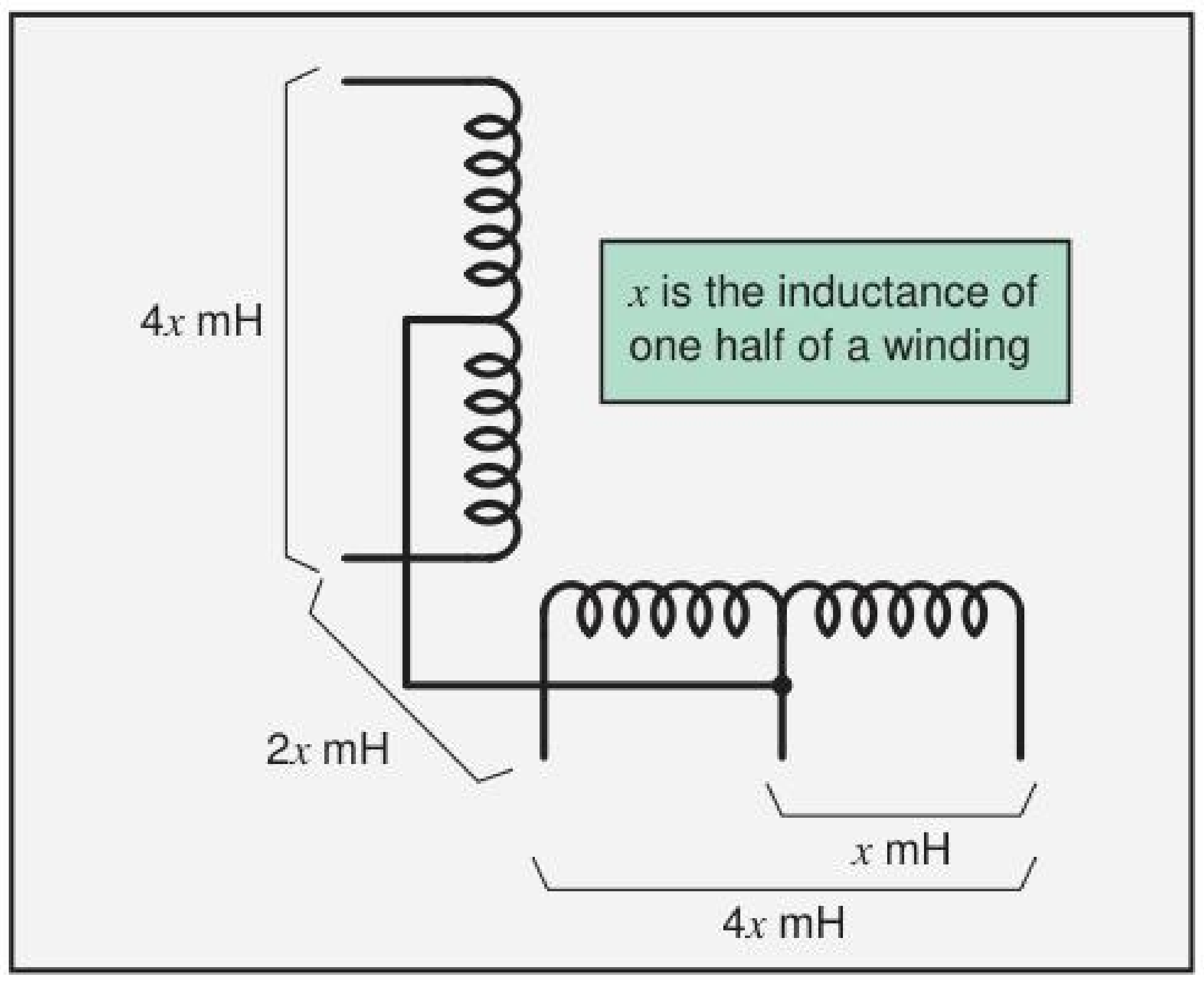


Fig.14. While the winding resistance will be the same, the inductance measurement will indicate which winding is which.

If you get a low resistance reading on all four leads then you have a VR motor.

5-lead stepper motors

This should mean it is a 4-phase unipolar motor but could be a 2-phase bipolar machine with an extra lead to the casing. Let's eliminate that possibility first, connect the red test lead to the motor casing and check the resistance to all 5 leads, if one wire gives near zero ohms and the rest open circuit then you have a 2-phase bipolar with a screen lead. Ignore the screen lead and check the remaining leads as described for '4 leads'.

It is more likely that this is a unipolar motor as per Fig.7 b). Since both pairs of windings are interconnected by the centre tap, this is a little more difficult to identify and will require you to measure and note down the resistance values as accurately as possible. There are 10 lead pair permutations you need to measure; four of them will be half the reading of the other 6. The four results with the half readings will have one wire colour in common; this wire is the centre tap. By way of an example, Table 2 is a representative set of results.

Identifying the centre tap is the most important for connecting to the driver, the rest can be found by trial and error as only one combination will cause the motor to step correctly. No harm will come to the driver or motor while trying out the permutations.

I have read of a technique using a battery or power supply to see which way a motor 'twitches' when DC power is applied between the centre tap and each of the other coil end leads in turn, but did not find the results reliable or convincing.

The windings are not only resistive, but also inductive, so I wondered if measuring inductances could be adopted. If you have an LCR meter (see Fig.13) then easy; if not, a scope can be used with a signal generator. You can purchase digital multimeters with inductance ranges for under £20, well worth the investment if you have a number of stepper motors to identify.

Measure the inductance of the six combinations of the remaining four wires (ignore the centre tap) (Fig.12)

See Table 3 for the measured inductances from the example motor and note that two readings were twice the value of the others. We know that each measurement is of two halves of a winding in series, and inductors in series have their values added together. So how can two readings be 4 times that of half a winding? The answer looks to be actually straightforward. In a stepper motor, the two individual windings or phases do not share a common magnetic flux path, but each individual centre tapped winding does. Each half winding of the motor under test had an inductance of 42mH, so two half windings in series gives 85mH and if they are on the same winding, the value is doubled again due to the common magnetic flux path. This doubling again does not happen with separate windings (Fig.14).

We now know which lead belongs to each centre tapped winding. From our test motor, red is the centre tap, pink/orange is one winding's ends and yellow/blue

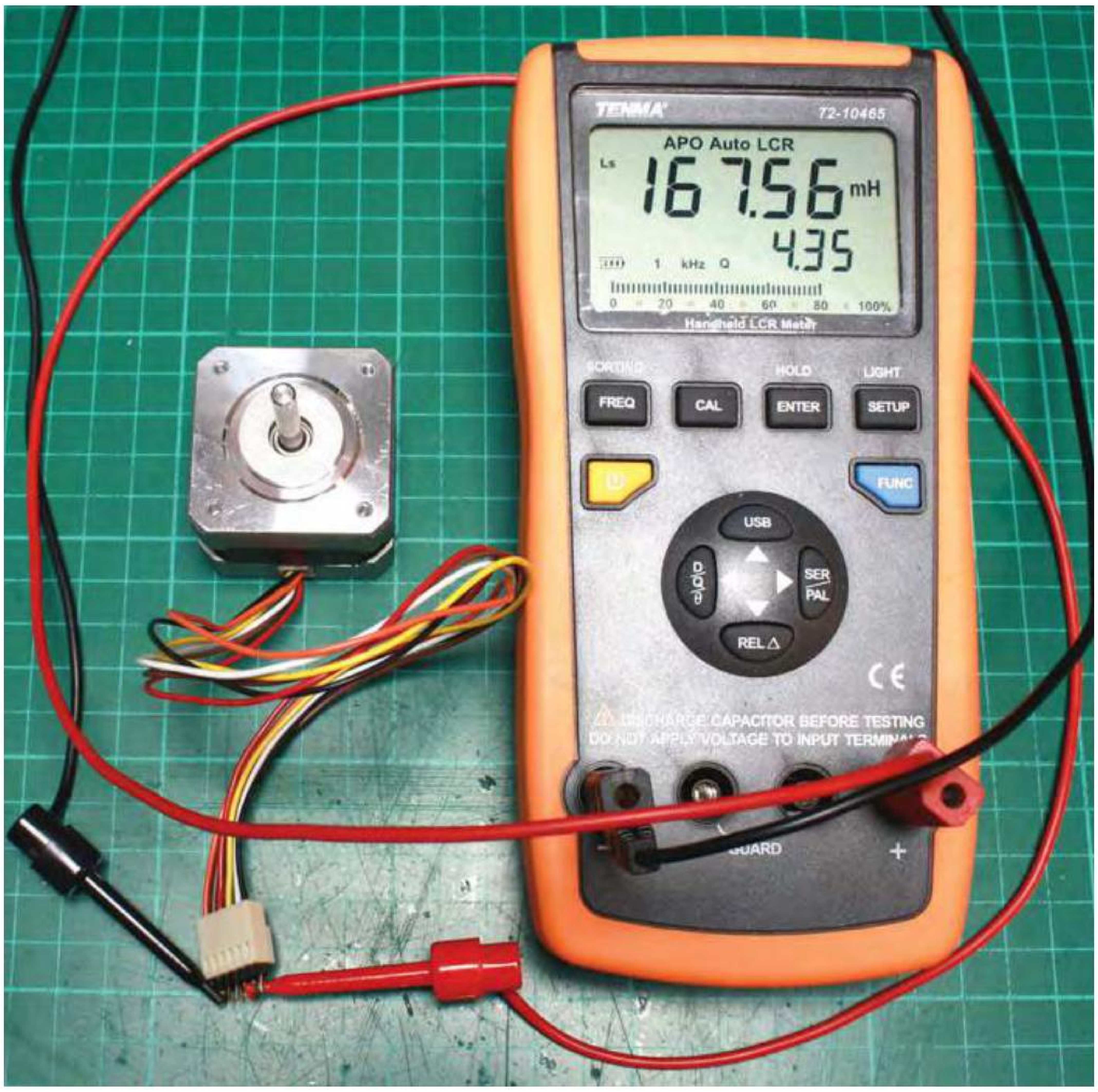


Fig.13. Using a LCR meter to measure stepper motor winding inductances.

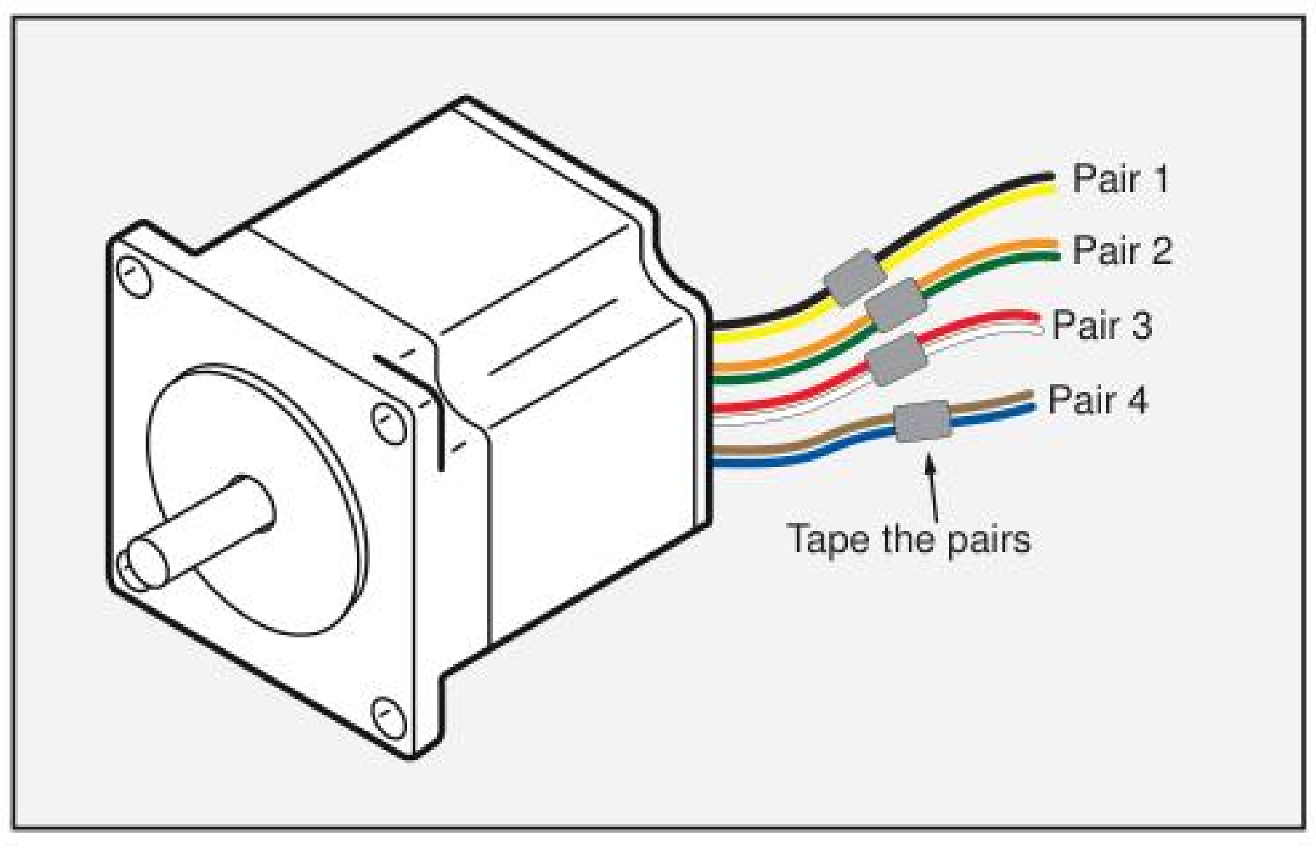


Fig.15. Find and group the 4 pairs of leads that give a low resistance reading.

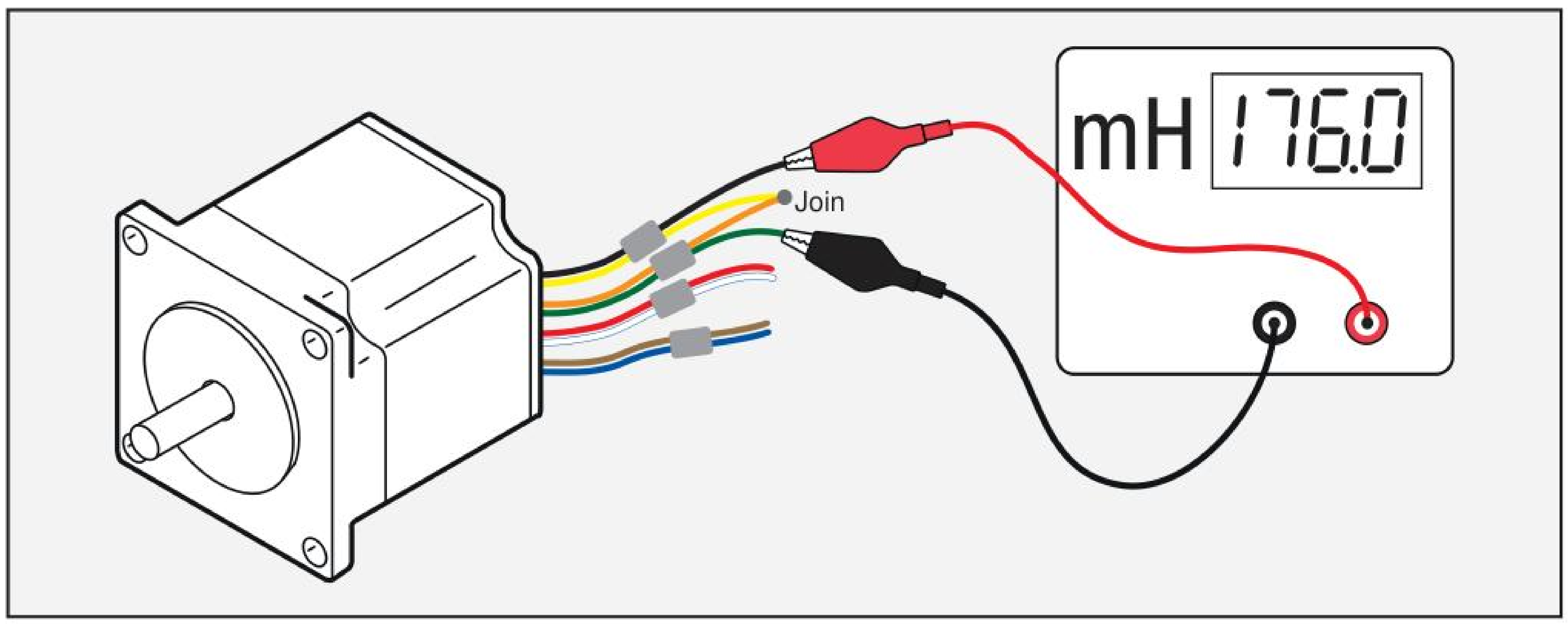


Fig.16. Measure the inductance to see which two pairs are on the same common magnetic flux path.

Table 3: In this example, two inductance values are twice that of the others.

Lead colour	Lead colour	Resistance (Ω)
Pink	Orange	167
Pink	Yellow	85
Pink	Blue	85
Orange	Yellow	85
Orange	Blue	85
Yellow	Blue	167

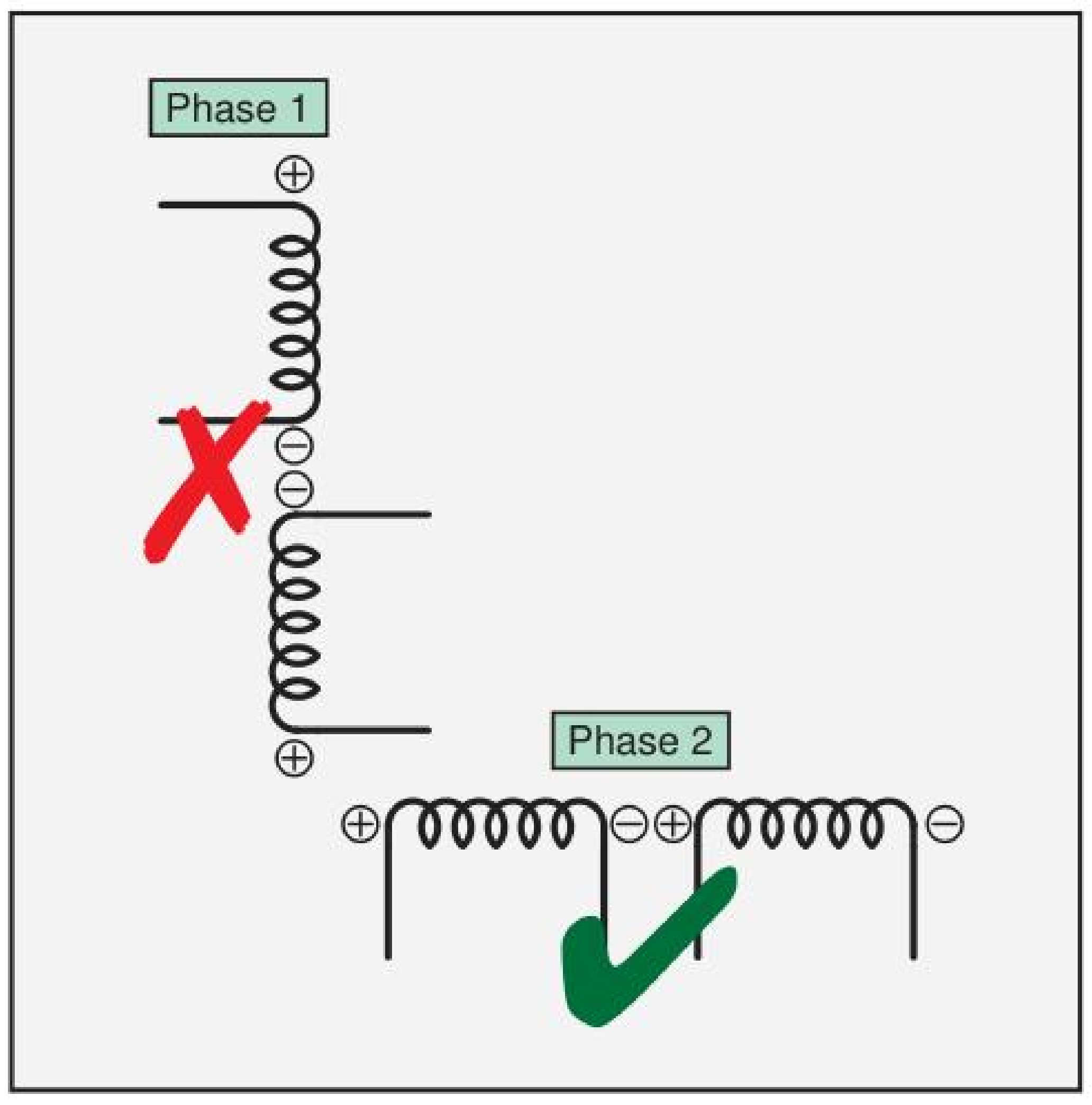


Fig.17. The polarity of the windings needs to be correct, here phase 1 is shown incorrect, it needs to be like phase 2.

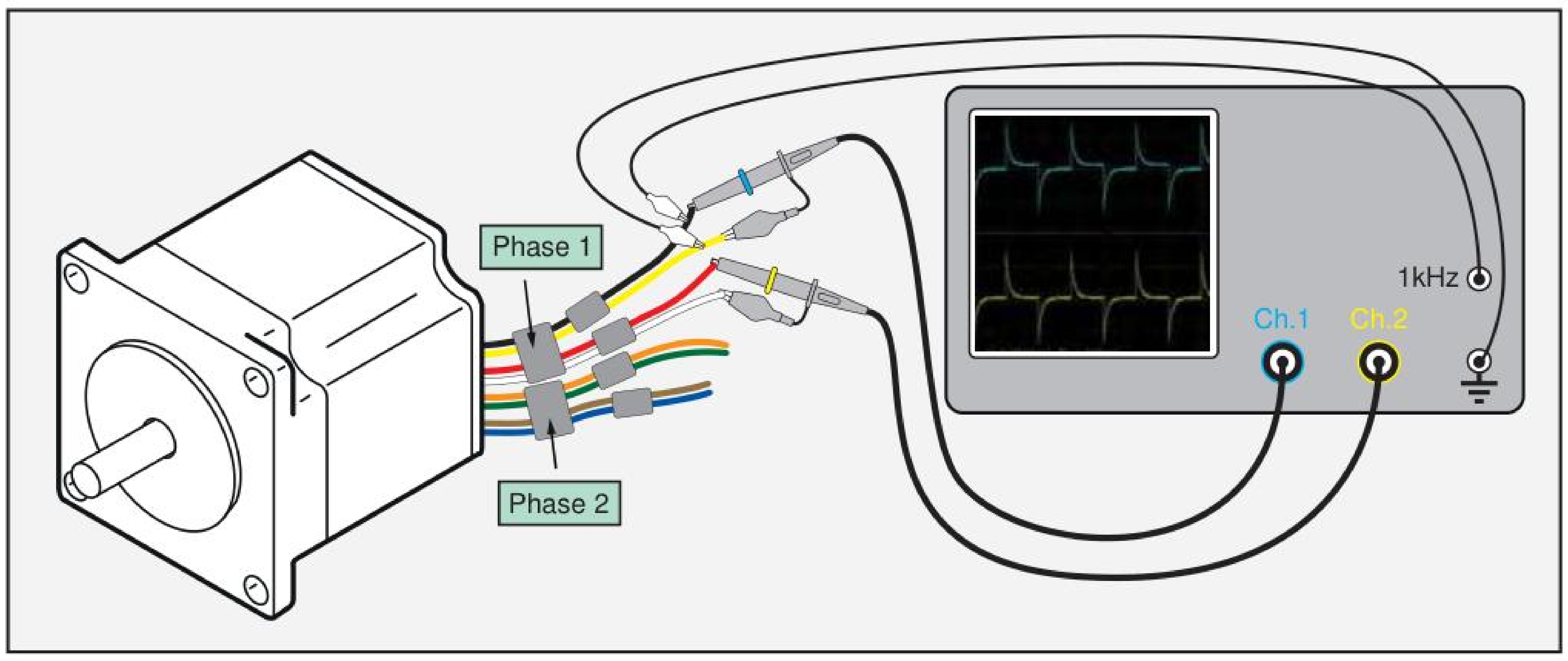


Fig.18. Inject a 1kHz signal into 1 pair and measure the induced voltage on an oscilloscope from the other pair of the same phase.

is the other. Now when you connect it to the unipolar driver, it will either work or not work. If it does not work, simply swap over one of the pairs of winding ends; ie, the pink and orange.

6-lead stepper motors

Unlike the 5-lead motors, the 6-lead motor is not a problem as the centre taps have been brought out separately as in Fig.7c.

As the two windings are not electrically connected to each other, you are looking for two identical groups of resistance readings. Clip you red multimeter test lead onto any lead and test the resistance of the other 5 leads, three of them are open circuit so put them aside for the moment. The two that gave readings will either be the same value or one twice the value of the other. If both readings were the same value then the red test lead is connected to the centre tap and the other two leads are the winding ends. If twice the value, move the red test lead to one of the two leads under test and repeat the measurements. If the readings are now the same, then the red test lead is connected to the centre tap. If still twice the value then the remaining lead must be the centre tap, try it and make sure. Repeat this for the other three leads put aside earlier until you have identified the centre tap. You now know the leads for each winding and which leads are the centre tap. Now when you connect the motor to the unipolar driver, if it doesn't step correctly, like for the 5-wire motor description above, swap over one pair of winding ends. As this is a 6-lead motor, you can drive it with a bipolar motor driver. For bipolar, do not connect the centre taps (cut off or individually insulate them), just the winding ends. If the motor steps in the wrong direct, as before just swap over one of the pairs.

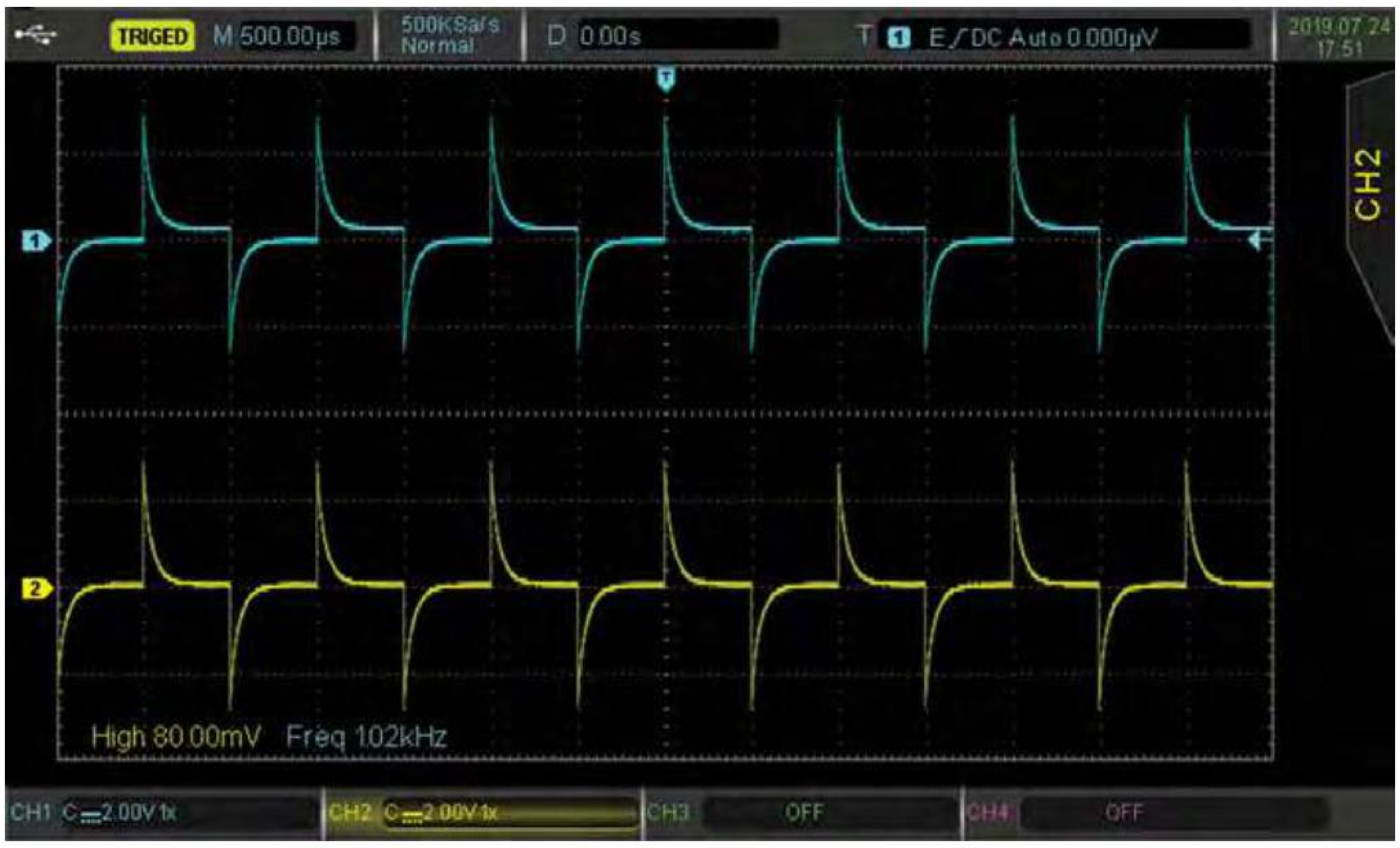
8-lead stepper motors

Now we really have a problem in that we have 4 individual windings and 28 possible measurements. Really you need the manufacturers data sheet, but let's see how far we can get by testing!

Carry out resistance tests to identify and mark the 4 pairs that each gives a low resistance reading, which we can call pairs 1 to 4 (Fig.15).

Measure the inductance of 1 pair which we can use to see which pairs are on the same common magnetic flux path by using inductance measurements as described

Fig.19. The square wave oscilloscope signal (channel 1) is applied to one pair in phase 1, channel 2 shows the induced signal into the second pair of phase one, the two winding pairs are in sync which is correct.



in the 5-lead motor tests. Connect pairs 1 and 2 in series and measure the combined inductance of pairs 1+2 (Fig.16).

If the inductance is 4 times that of the individual pair we measured then we know the both of these windings are on the same magnetic flux path which we can call phase 1, pairs 3 and 4 must then be phase 2.

If the combined inductance reading is only twice that of a single pair then series connect pairs 1 and 3, and again measure the combined inductance. If still only twice the reading try pairs 1 and 4, one of these combinations should give the four-times reading we are looking for.

We now know which pairs are in which phase, so best to group the pairs into their phases. What we do not know is which way around the two windings in each phase are oriented (Fig.17). This is important because when we connect an 8-lead motor to a driver, the phase windings are combined in either series or parallel. Resistive or inductive measurements will not help determine the polarity of the windings, for that you need an oscilloscope.

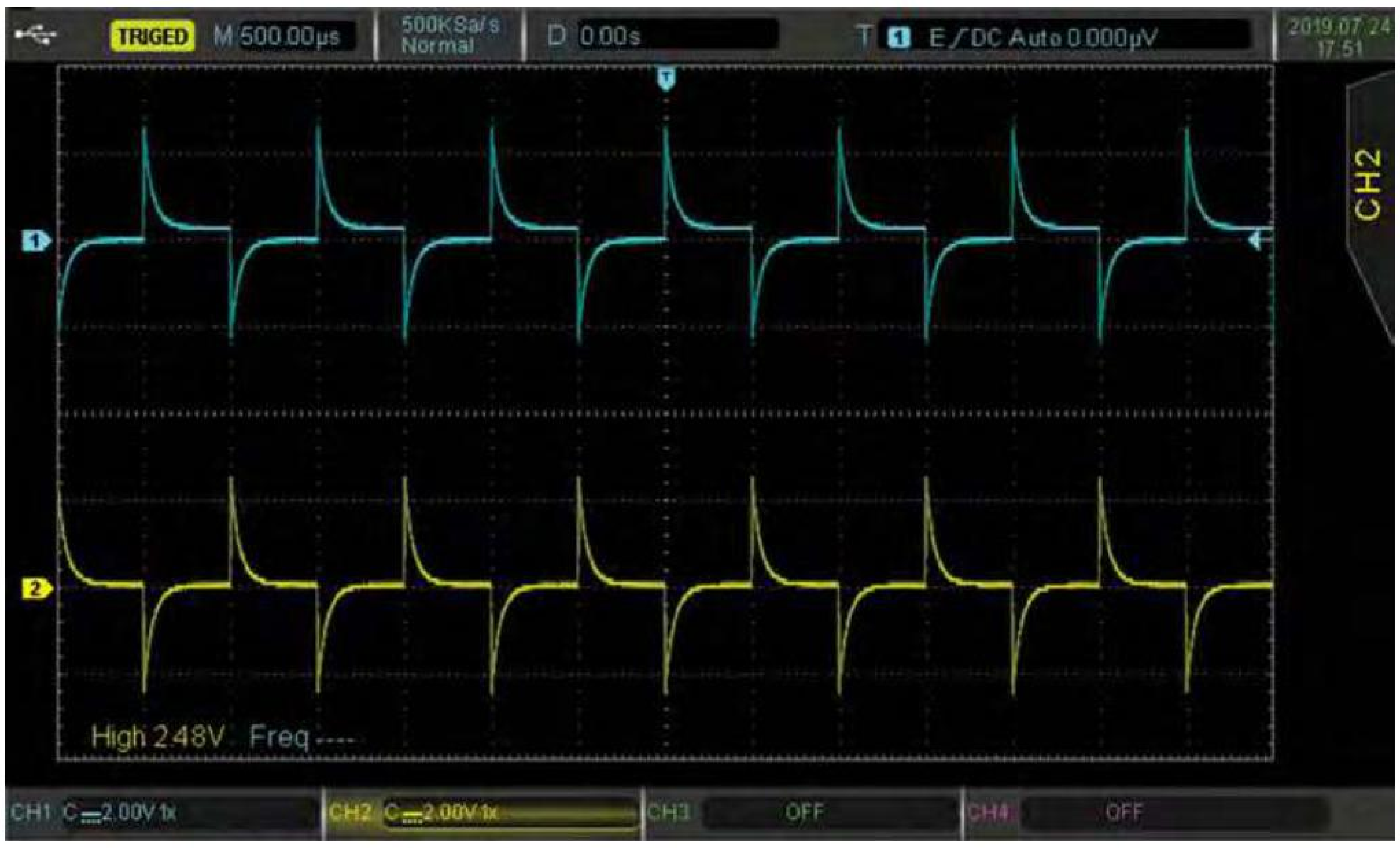


Fig.20. If the two windings are out of phase, the measured signal on channel 2 is phase shifted by 180°.

Oscilloscopes usually have a 1kHz signal source used for trimming the capacitance of the scope probes, but we can use it as a signal source. Connect the scope's 1kHz signal source to one of the winding pairs in phase 1 and to channel 1 on the oscilloscope. Now connect the other phase 1 winding pair to channel 2 on the oscilloscope. If the two waveforms are in sync, as in Fig.19, then you can deduce that the signal source ground and channel 2 ground connections go to the same ends of the two windings, this is better illustrated in Fig.18. If the two waveforms are 180° out of sync (Fig.20), then swap over the channel 2 scope leads at the motor and the waveforms will then be in sync.

Repeat the same test for phase 2 windings and you will have identified the windings sufficiently to connect to your driver.

Next month

In Part 3, we will look at how to drive stepper motors.

Technobots
Supporting your projects since 2001

Mechanical
Bearings, Gears, Sprockets, Pulleys, Shaft, T-slot extrusion

Motors
Stepper motor drivers, DC controllers, Miniature and Geared Motors

Electronics
Passives, semiconductors, breakout boards, sensors, connectors and leads

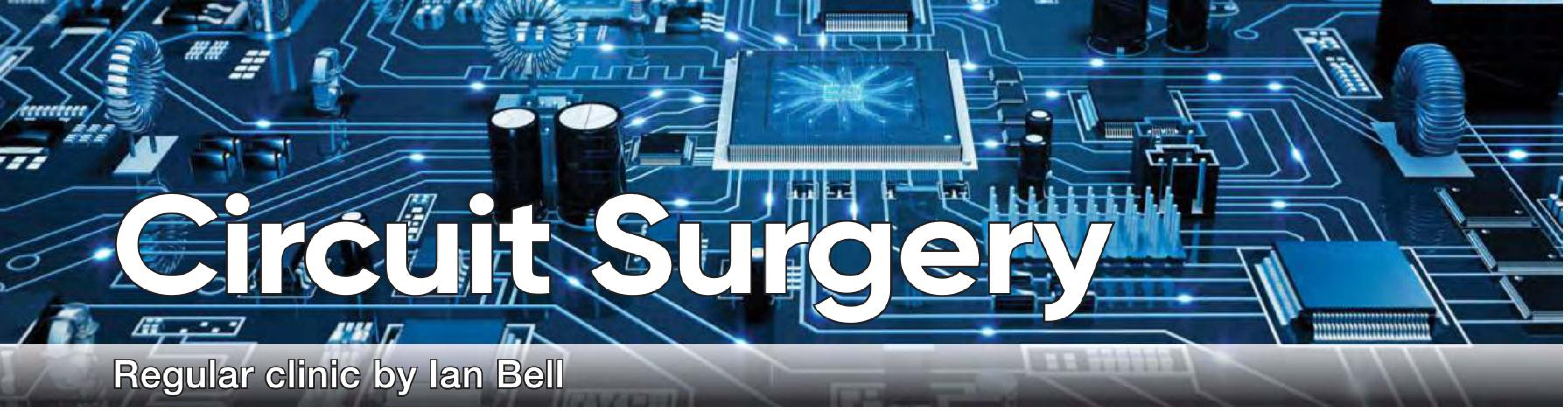
MakerBeam
Pololu Robotics & Electronics
FingerTech Robotics
MFA/Como Drills
MaxBotix

Order online at
www.technobotsonline.com

Follow us at
facebook.com/TechnobotsUK

Unit 29, Webb Ellis Business Park
Rugby, Warwickshire
CV21 2NP

Tel: 01788 877 400



Circuit Surgery

Regular clinic by Ian Bell

Strain gauges and differential signals

Recently, Scott Siler posted a question on the EEWeb forum about a problem he was having creating an amplifier for a strain gauge.

'I'm an aerospace engineer working on a side project to gather data from a strain gauge using an NI DAQ (a National Instruments data acquisition device). The output is 0-36mV DC so I need a gain of 3 to better use the DAQ. My EE experience is limited to a couple of classes I took during my bachelor's. I have an AD8628 op amp and built a basic, non-inverting negative-feedback circuit, as shown in the attached picture (Fig.1). The rail voltage is 0-3V DC. It does not seem to have any gain, but instead the output voltage is actually lower than the input. I'm sure I must have wired it incorrectly, but I cannot tell how. Any help would be greatly appreciated.'

The initial post lacks some details, but as part of the discussion in response to his post, Scott added some more information. He is using a commercial load cell rather than a basic strain gauge foil (see below for an explanation of this) with a 12V DC excitation voltage. The load cell has a basic output of 0-36mV, which Scott is trying to increase to 0-100mV in order better use the available resolution of the recording capabilities of the DAQ. He is using the differential input of the DAQ, which has a resolution of 18 bits. We do not have full details of the load cell.

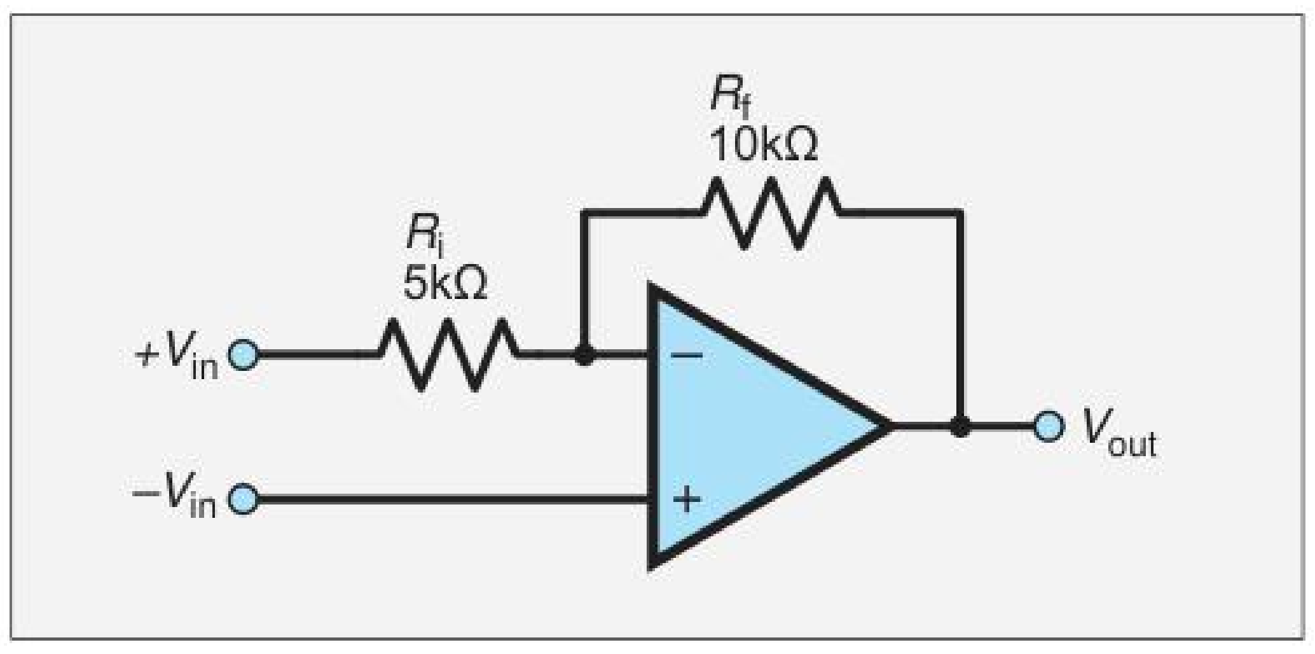


Fig.1. Scott's circuit.

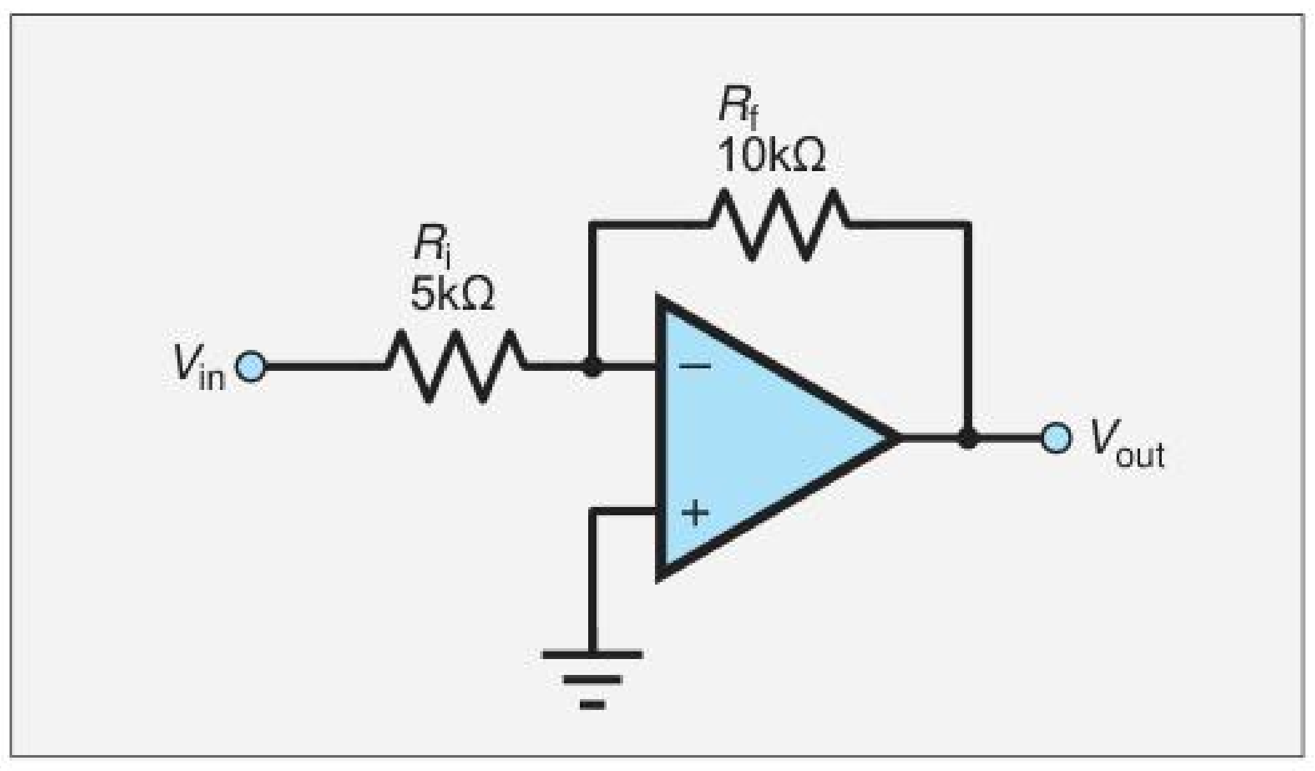


Fig.2. Standard op amp inverting amplifier.

A number of issues are raised here: strain gauges and how to use them, the sources of errors in such circuits (and sensor signal conditioning in general) and related design trade-offs. However, perhaps the most fundamental issue, which was quickly pointed out in the forum discussion, is that the load cell output is a *differential* signal, and therefore needs a differential amplifier – the circuit in Fig.1 is not suitable. The circuit in Fig.1 is based on the standard inverting op amp amplifier, which usually has a single input (via the R_i resistor) with the non-inverting input either grounded (as in Fig.2) or connected to another fixed reference voltage. The inputs are not symmetrical and will interact differently with external circuitry, particularly resistor networks, which we assume are present in the load cell (see later discussion).

This month, we will look briefly at strain gauges and then discuss differential signals and how to amplify them in general.

Strain gauges

Strain is a measure of the deformation (change in size or shape) of an object – the change in location of specified points relative to a reference length. Strain can occur due to applied force or a change in temperature of the object. A strain gauge is a sensor whose electrical resistance changes with deformation. It typically comprises a thin, flexible, insulating foil, which supports a long conductive strip, typically in a zig-zag pattern (see Fig.3). At the two ends of the conductive strip there are contacts to connect the device to the measurement circuit. A typical resistance value is 120Ω.

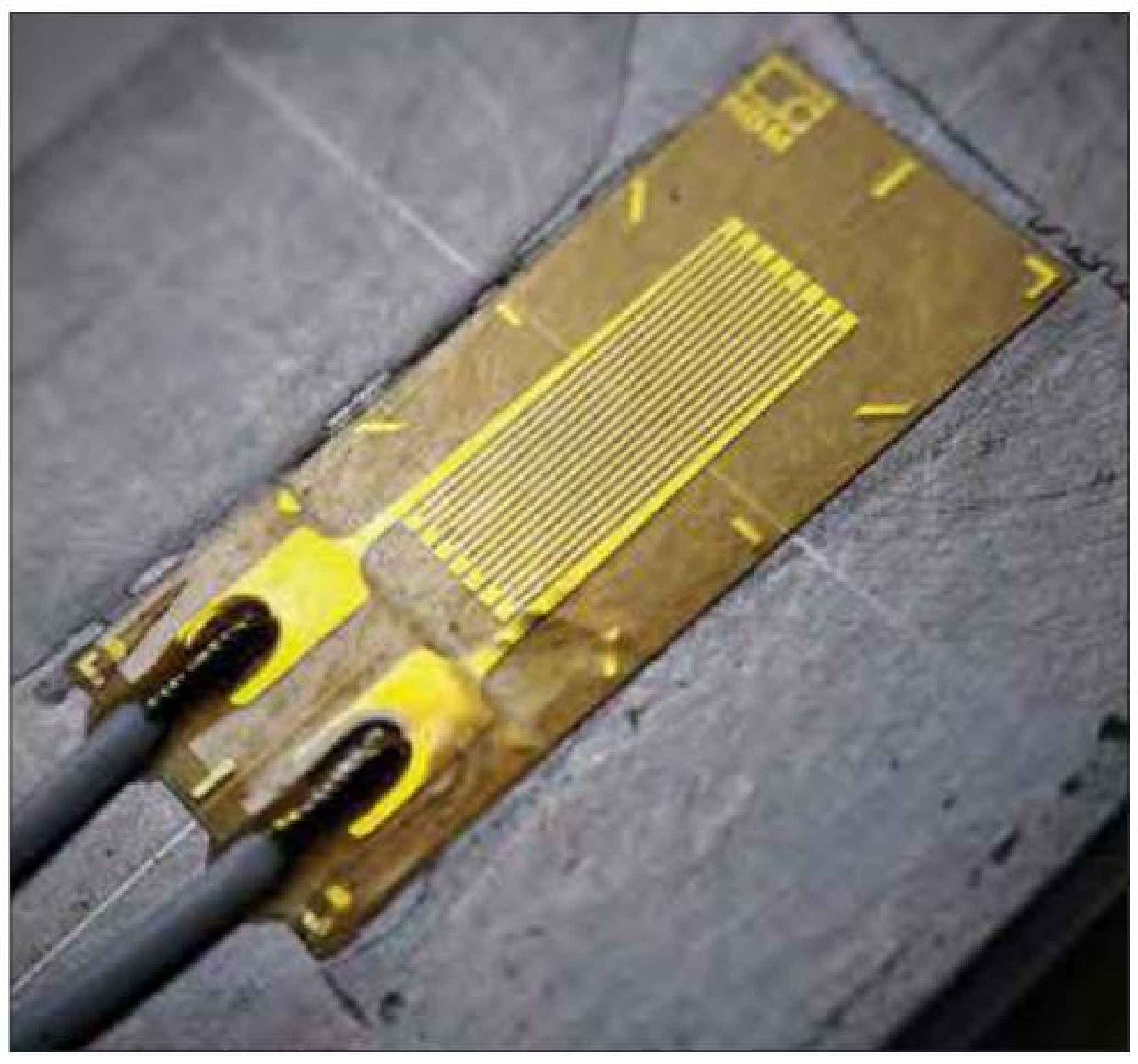


Fig.3. Typical strain gauge (Wikimedia Commons license, Cristian V).

When a strain gauge is attached to an object it will deform with it, enabling the object's deformation to be measured. The attachment must be done correctly (eg, with a suitable adhesive to a compatible material) in order for the gauge to work correctly. Strain gauges can be used in their basic form to measure object deformation directly, for example to study a structure, or they can be built into a larger device, called a 'load cell'. In a load cell, the strain gauge is attached to a specifically

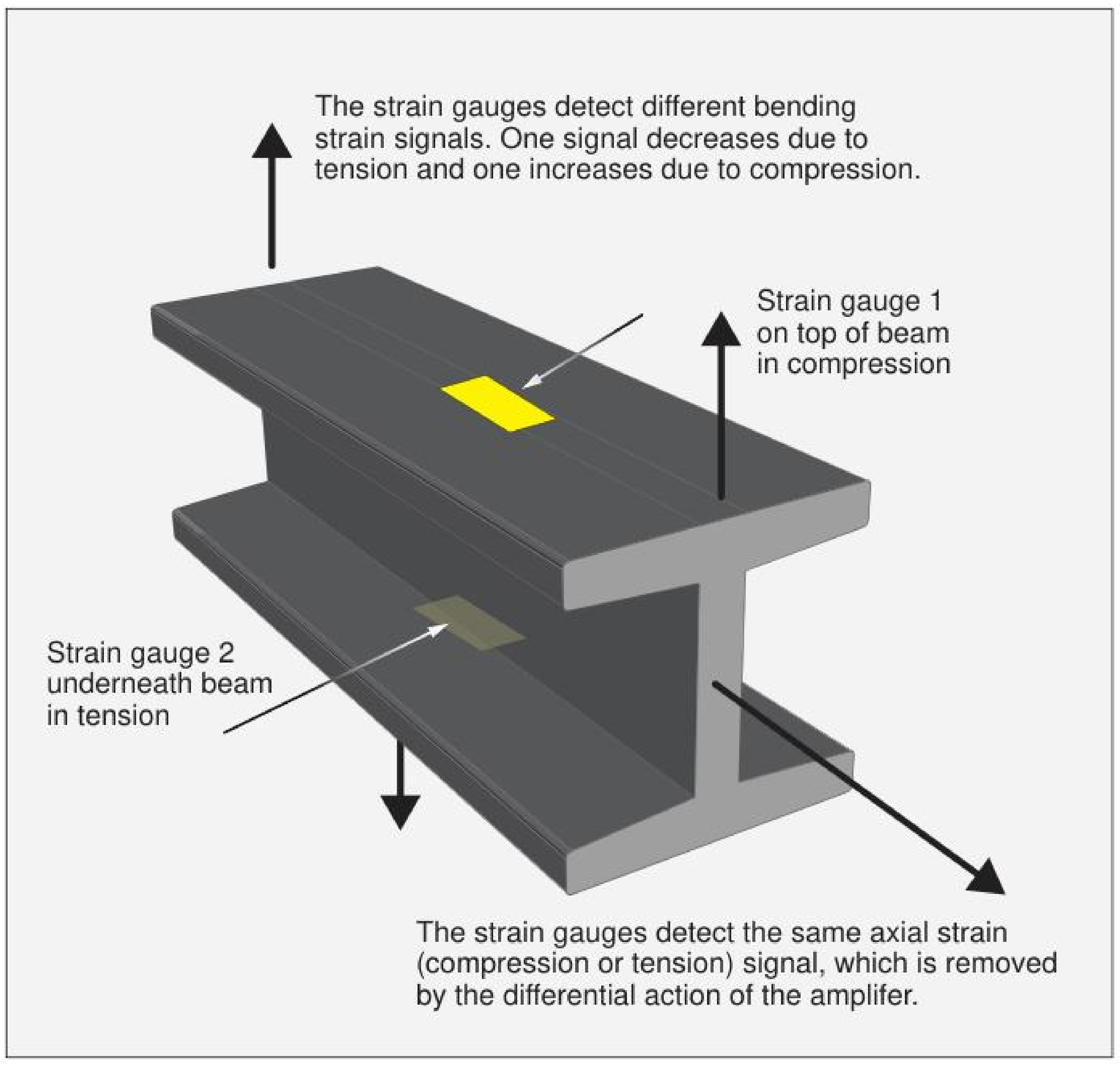

The strain gauges detect different bending strain signals. One signal decreases due to tension and one increases due to compression.
Strain gauge 1 on top of beam in compression
Strain gauge 2 underneath beam in tension
The strain gauges detect the same axial strain (compression or tension) signal, which is removed by the differential action of the amplifer.

Fig.4. Strain gauges mounted on opposite sides of a beam to measure bending under load. Axial stretching or compression (along the beam) is ignored because the gauges generate identical signals which cancel eachother in the differential amplifier.

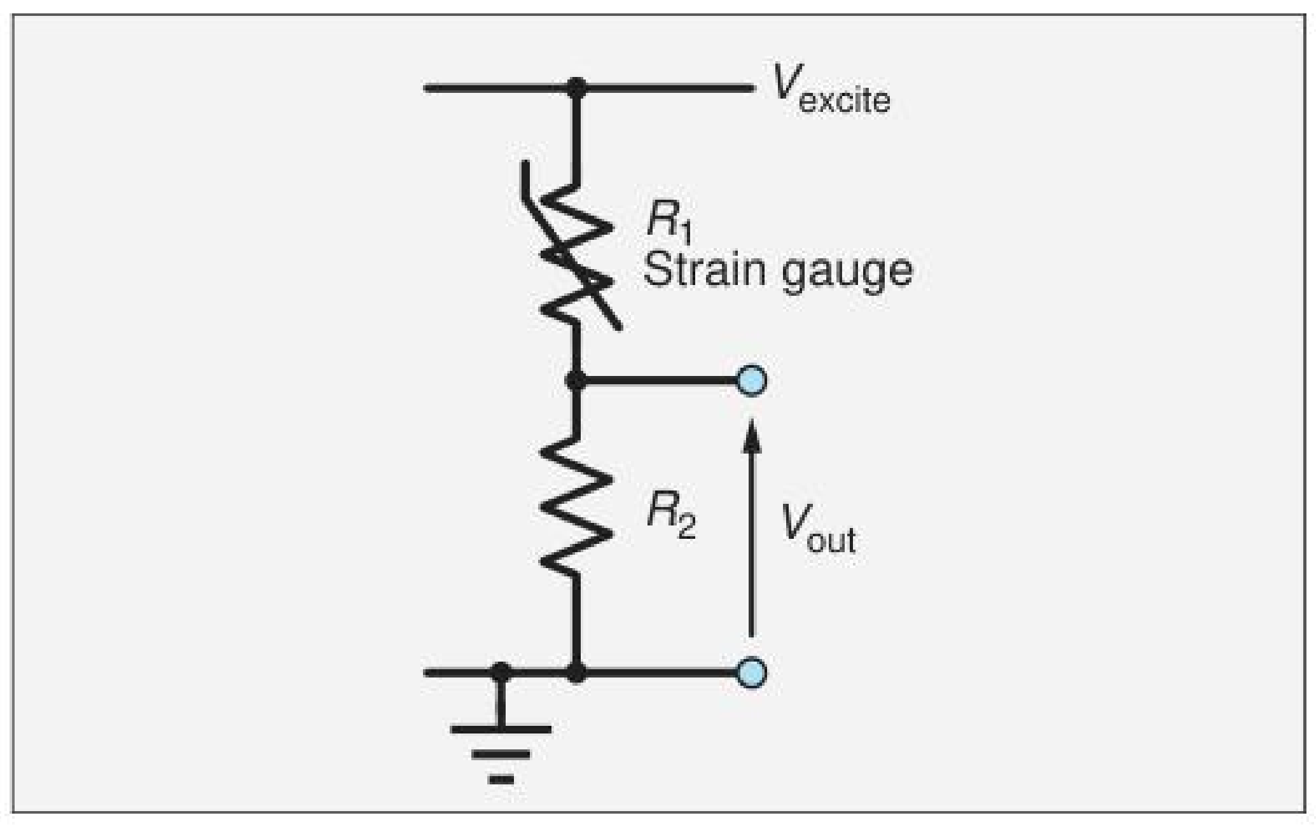


Fig.5. Strain gauge in a potential divider.

designed metal body which deforms when force is applied to the device. Load cells are much easier to use than strain gauge foils and have many industrial applications in force and weight measurement.

Multiple strain gauges are often used together in variety of different arrangements. For example, placing two strain gauges on the opposite sides of a bar or beam (see Fig.4) provides a way of measuring strain to due bending of the beam. As the beam bends, one strain gauge will stretch and the other will compress. This will produce opposite changes of resistances, yielding a stronger signal than a single strain gauge. Furthermore, if the beam is also subject to compression or tension along its axis, then the strain gauges will respond equally and this effect can be ignored by a circuit which only responds to opposite changes in the two strain gauges – thus it is just the bending which is measured. Similarly, changes in temperature, which are potentially problematic if a single strain gauge is used, will usually effect both equally and will not affect the output from a suitable circuit.

To measure a resistance (obtain a measurement signal from the strain gauge) some current must be passed through it. A simply way to achieve this is to use a potential divider, as shown in Fig.5 – an excitation voltage (V_{excite}) provides the current flow to facilitate measurement. The other divider resistor (R_2) is typically chosen to match the strain gauge's value at the mid-point of the measurement range. The output voltage of the potential divider could be fed directly to an analogue-to-digital converter (ADC), or buffered by an amplifier, as required.

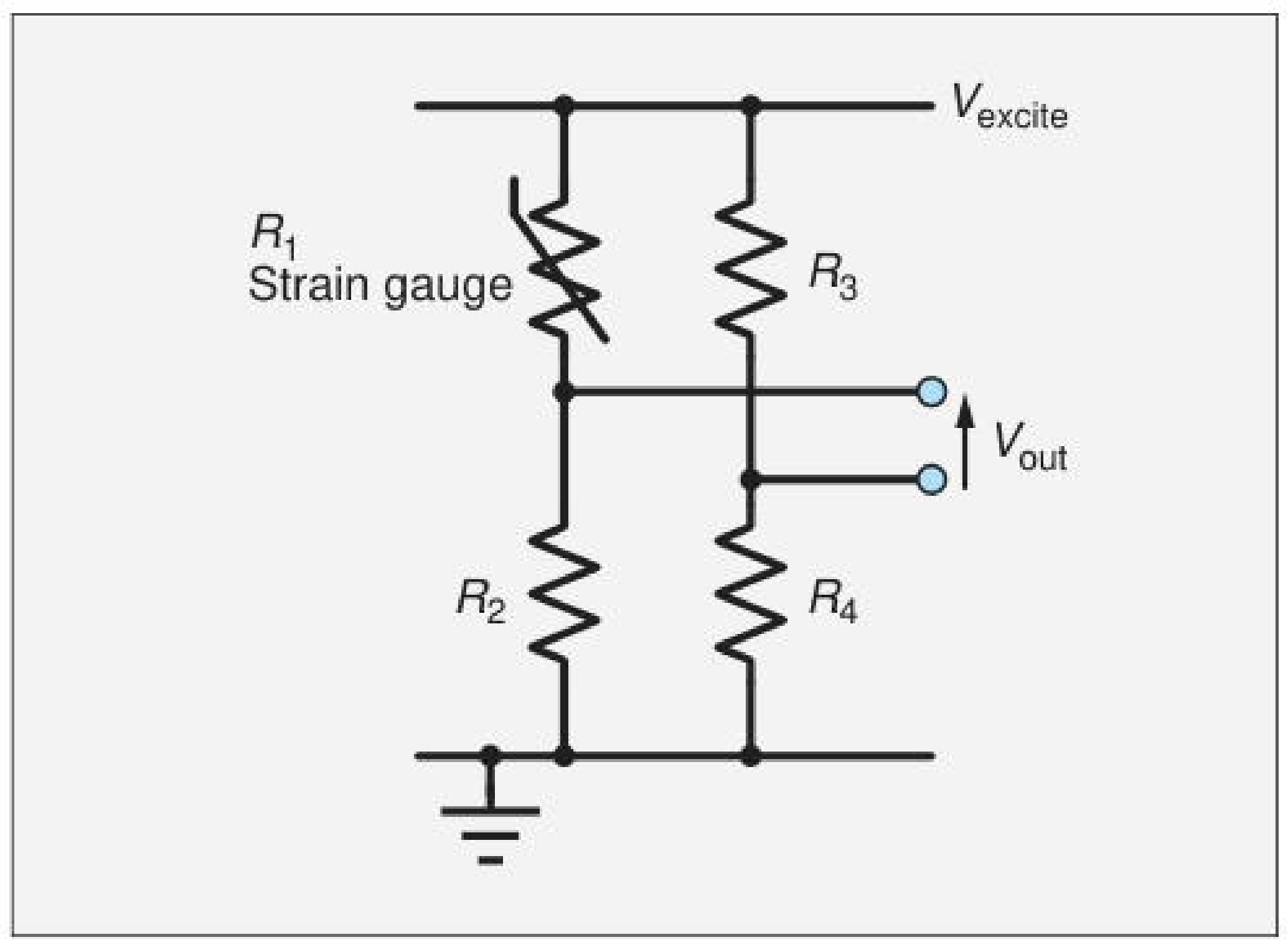


Fig.6. Strain gauge in a bridge circuit – in this example, R1 is the strain gauge, but various arrangements can be used.

Wheatstone bridge

A single potential divider that's used with a sensor that exhibits small resistance changes will produce an output voltage that only varies over a small range on top of a large offset. This can be difficult to use (eg, to digitise to useful resolution). A better approach, which facilitates higher accuracy and is employed in most strain gauge applications, is to use a Wheatstone bridge circuit, as shown in Fig.6. The bridge circuit is also often drawn with the layout shown in Fig.7. From Fig.6 it can be seen that the bridge is two potential dividers in parallel, with the output voltage being the *difference* between the two potential divider voltages. The differential voltage from a bridge does not have the offset associated with a simple potential divider, so it can easily be amplified (by a suitable differential-input amplifier) without the offset causing the amplifier to saturate.

The circuit in Fig.6 has a single strain gauge (R_1). The other three resistors will all typically match the strain gauge's resistance at some convenient point, in which case the output voltage will be zero at that point because both potential dividers will be producing the same voltage (and the output is the difference between them). The output can be negative or positive if the resistance of the strain gauge can change both up and down from the zero-output point set by the potential divider resistances. A negative output is not a voltage below ground – it is a negative difference between the two potential dividers, and whether you consider a given output negative or positive will depend on which way round you designate the two connections in terms of polarity.

The circuit in Fig.5 has a single strain gauge with three fixed resistors, but as we already noted, there are a variety of possible physical measurement scenarios (such as the bending beam example) which use multiple stain gauges. The bridge can contain one, two, three or four strain gauges depending on the setup used. In some situations, additional resistors may be used (eg, between the excitation voltage and bridge) to fine tune the behaviour of the circuit. And finally, in our quick look at these circuits, it should be noted that the excitation voltage plays an important role in the accuracy of the measurement – if it varies, so will the measured output. If a bridge circuit is used with an ADC the excitation voltage can be used as the ADC's reference voltage. This will compensate for any variation in excitation voltage. Alternatively, the excitation voltage can also be measured and compensation performed in software.

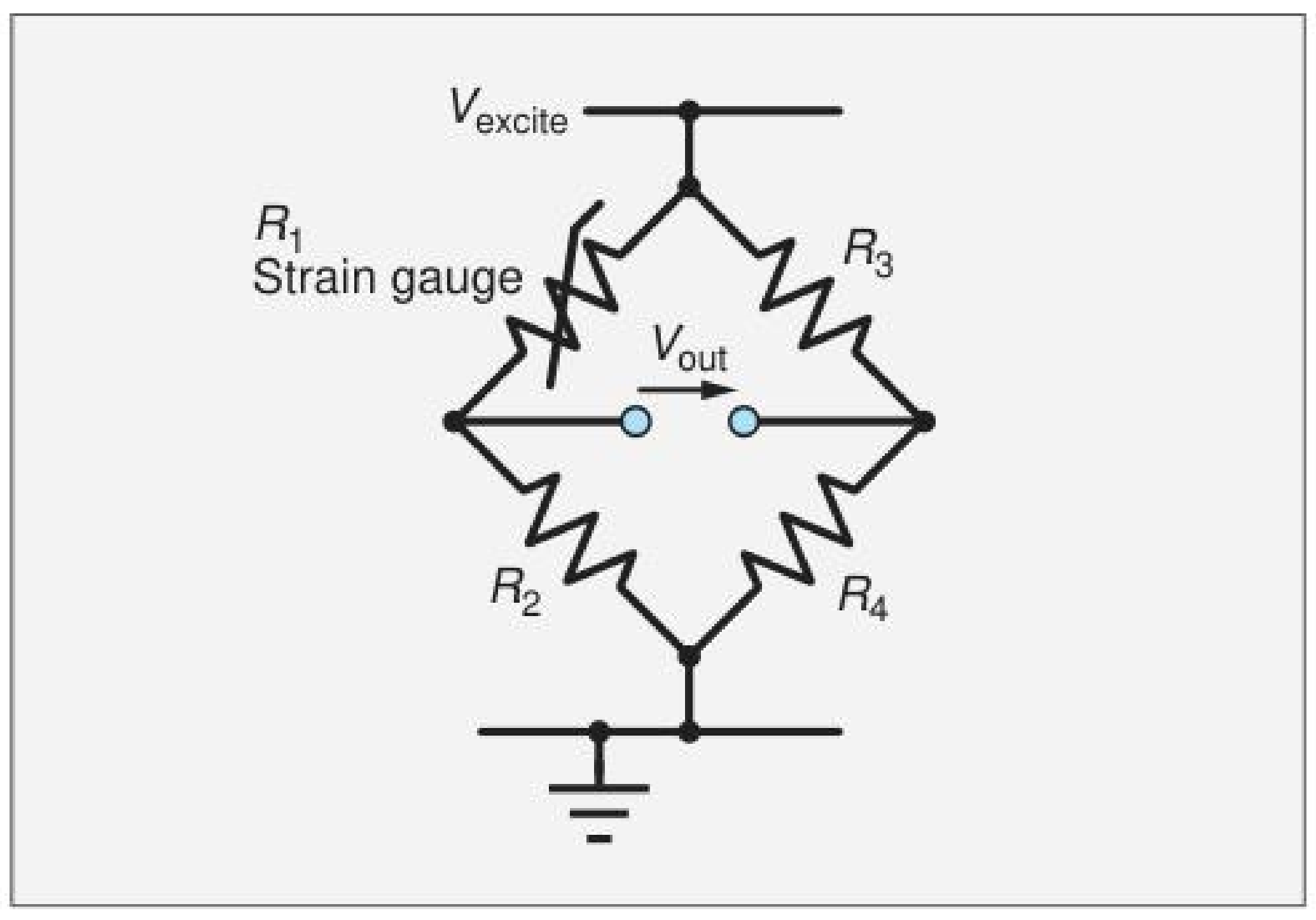


Fig.7. The same circuit as in Fig.6 drawn with an alternative layout.

Differential Signals

As noted, the output from the Wheatstone bridge circuit is a differential signal. In the rest of the article we will look at the basics of differential signals in general – they are very widely used in electronic systems, not just in strain gauge circuits. A differential signal is carried on two wires other than ground, so we can observe the voltage on each wire individually (eg, V_1 and V_2), but the actual signal is equal to the difference in the voltages between the two wires, each measured with respect to ground. So if the two voltages on the wires are V_1 and V_2, then the differential signal is ($V_1 - V_2$). Signals carried on just one wire are referred to as 'single ended' to distinguish them from differential ones.

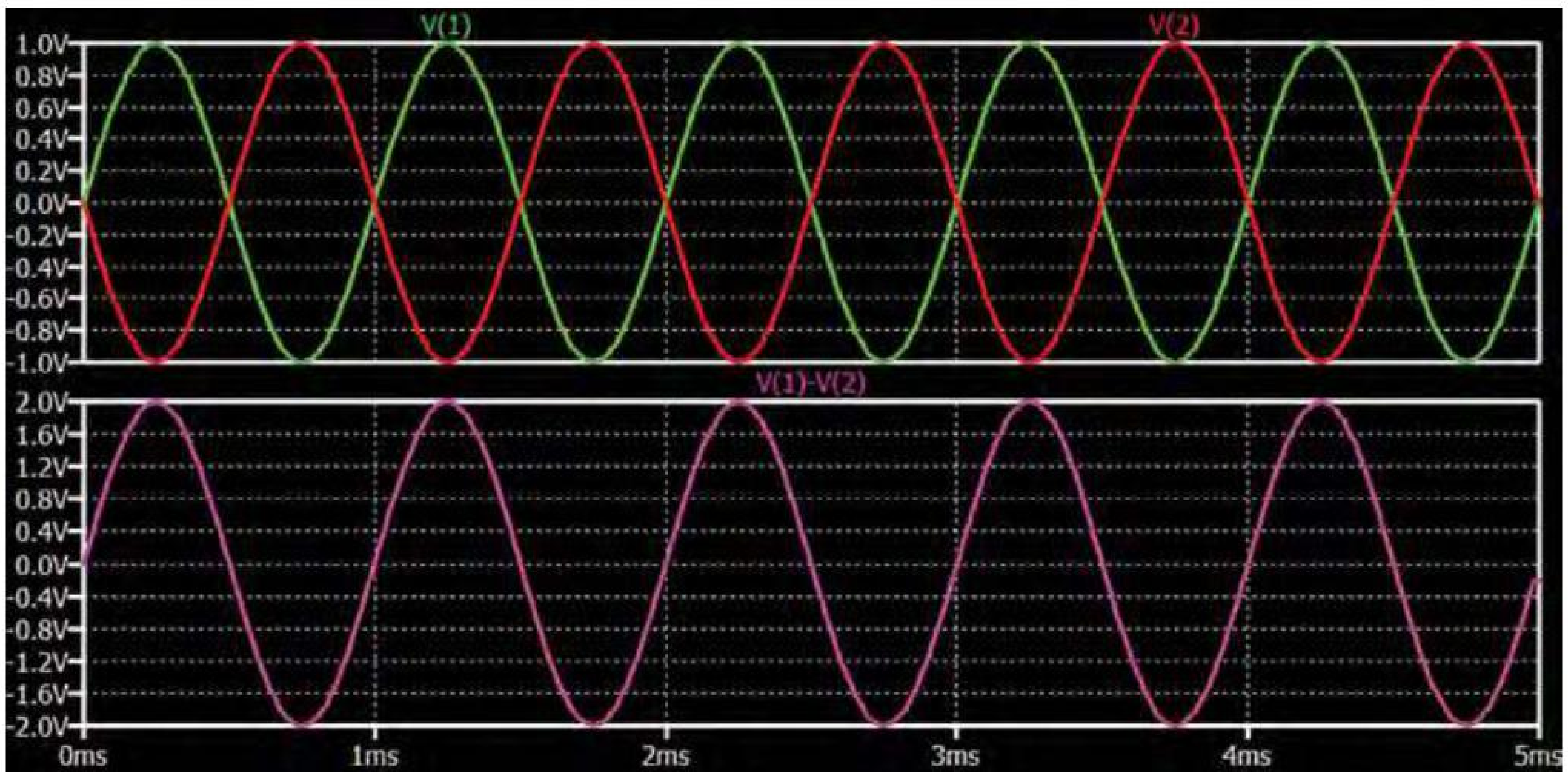


Fig.8. Differential signal: the voltages on the two individual wires V1 and V2 are shown in the upper plot. The actual signal is difference between V1 and V2 and in shown in the lower plot.

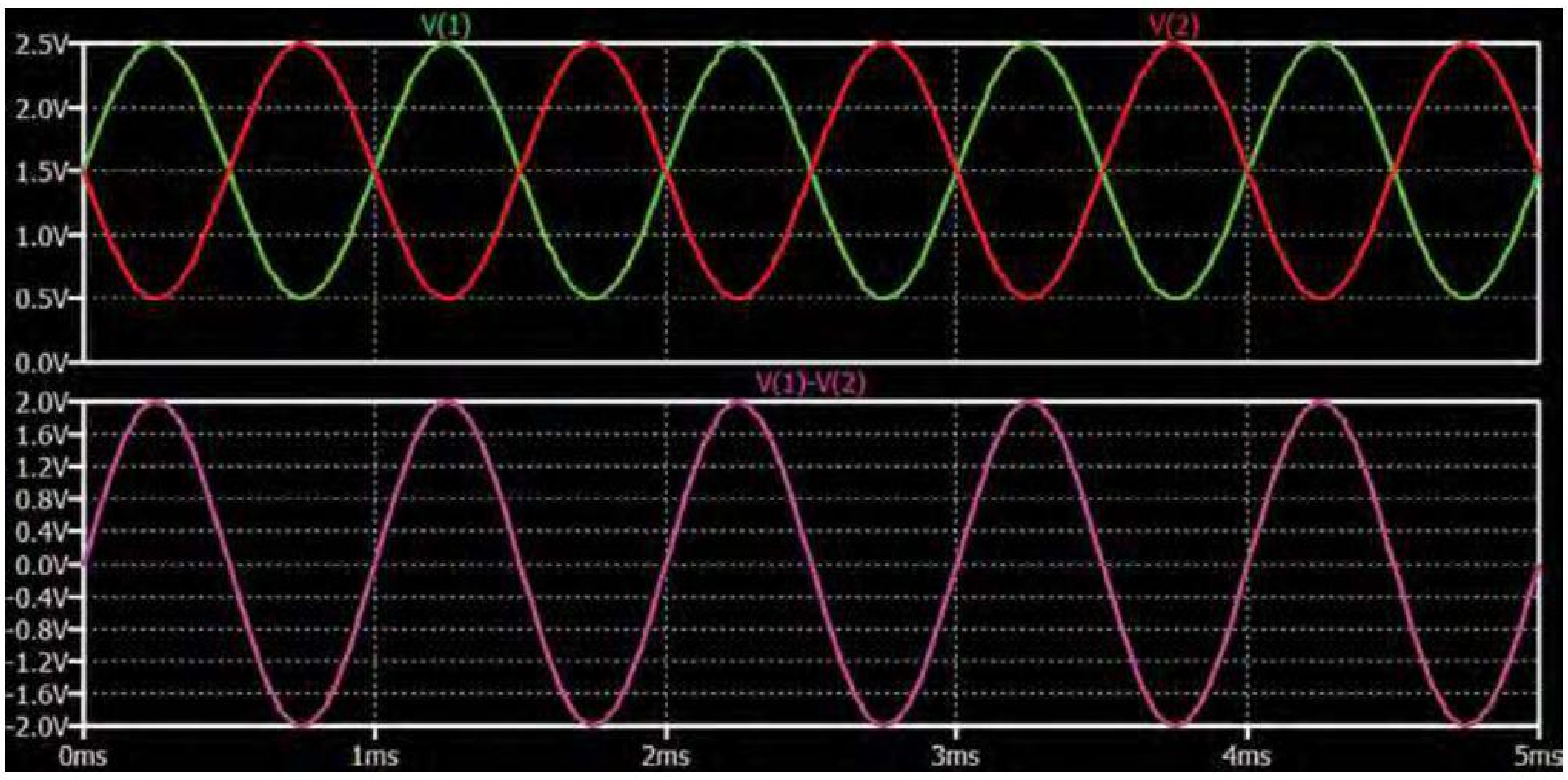


Fig.9. this is the same differential signal as shown in Fig.8, but with a different common mode component (1.5V DC in this case).

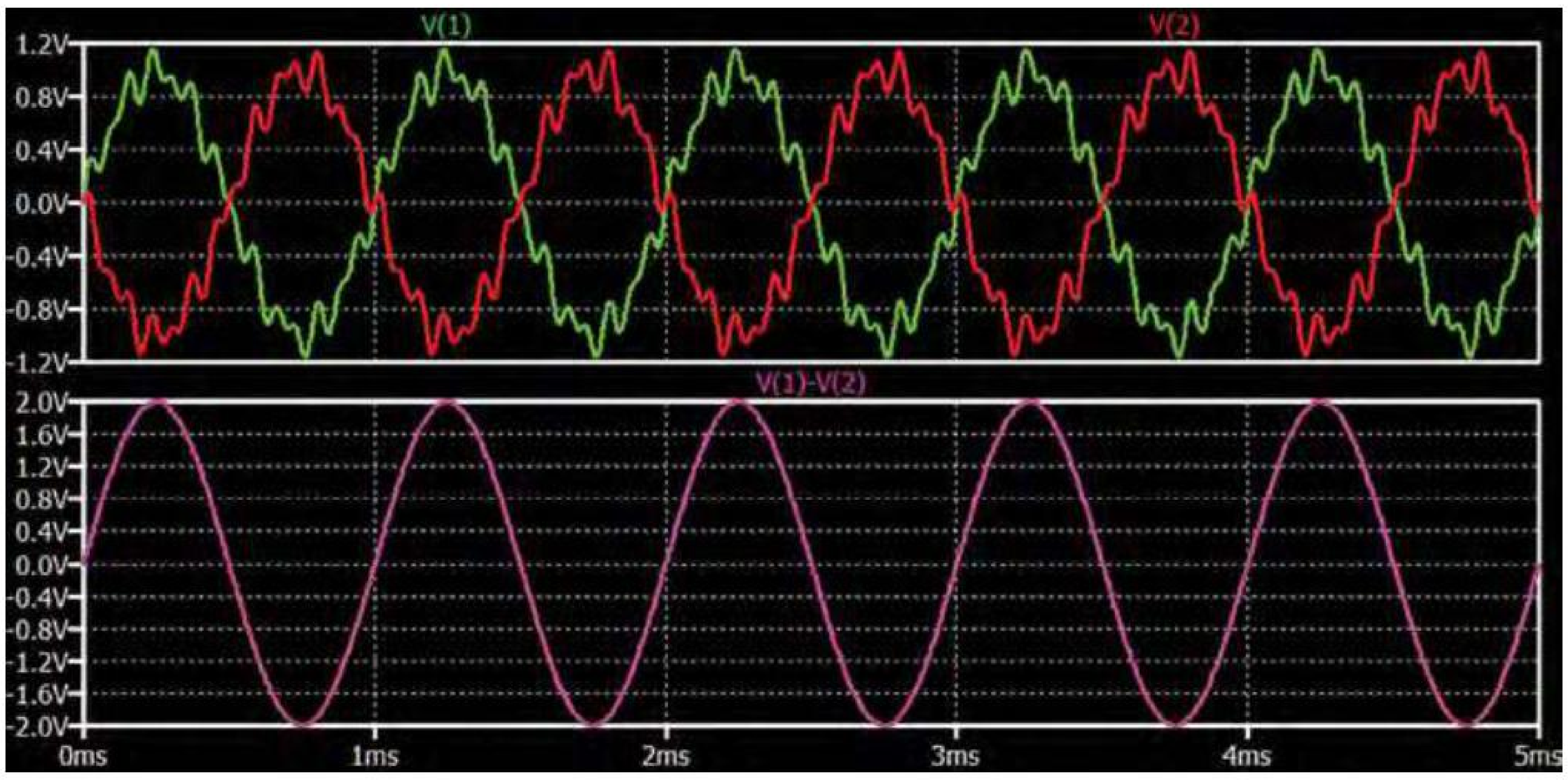


Fig.10. The same differential signal as shown in figures 8 and 9, but with a different common mode component (a higher frequency AC signal in this case).

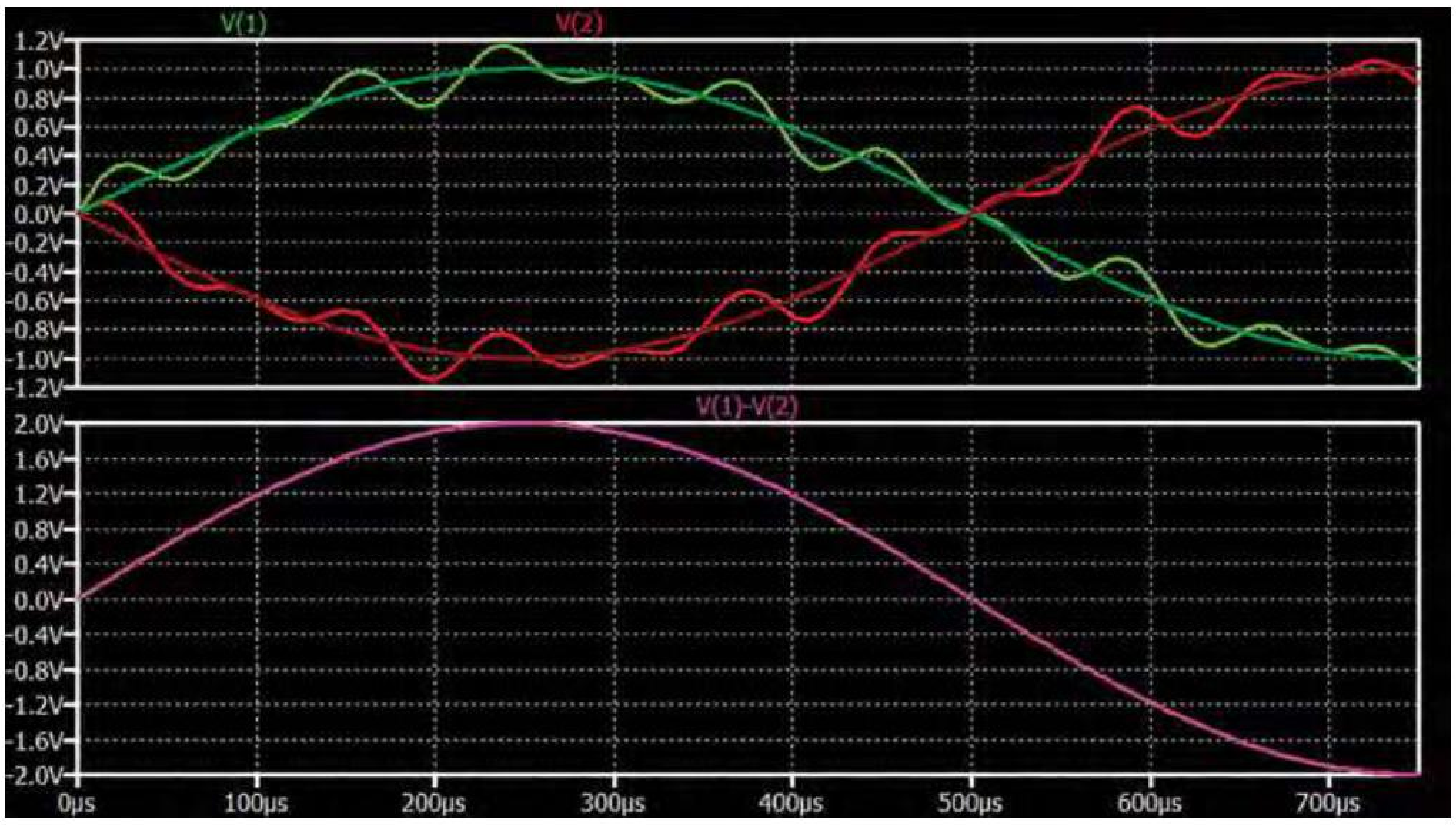


Fig.11. Zooming in on the first part of Fig.10 to see more clearly that the higher frequency signal is common mode – i.e. it is equal and in the same direction on both waveforms and hence the difference between these waveforms is the same as the signal in Fig.8 and Fig.9.

Fig.8 shows a symmetrical differential signal that is a 1kHz sinewave with a peak voltage of 2V (4V peak-to-peak). The two individual voltages (V_1 and V_2 on the two wires that carry the signal) are shown on the upper plot – these are equal and opposite and have peak voltages of 1V. Because they are opposite, the peak difference between them is 2V, which is the amplitude of the differential signal. The signal itself is shown on the lower plot – this is ($V_1 - V_2$).

Common mode

Another differential signal is shown in Fig.9 – this is also a 1kHz sinewave with a peak voltage of 2V. The lower plot (the differential signal) is exactly the same as the one in Fig.8, however, the signals on the individual wires are not the same. In Fig.8, both V_1 and V_2 are centred on 0V, whereas in Fig.9 they are centred on 1.5V. In order to fully describe a differential signal we need to state two things – the differential signal itself, which is the difference between the voltages on the two wires, and the voltage they have in common – called the 'common-mode' voltage. The common-mode voltage is the average of the voltage on the two wires.

Common-mode voltages do not have to be DC – they can be any signal form or wave shape. Fig.10 shows an example of exactly the same differential signal as in Fig.8 and Fig.9, but with a higher frequency AC common-mode signal. Fig.11 is a zoom-in to part of one cycle with the individual parts of the differential sinewave added to guide the eye in seeing that the higher frequency signal is in the same direction on both waveforms and hence is ignored or cancelled out when the difference is taken.

A common situation in electronic systems is to have a wanted differential signal with an unwanted common-mode component. An example of where this might occur is with a signal that is carried on long wires through an electrically noisy environment. If the two wires carrying the differential signal run closely together then external effects (eg, mains hum, radio interference…) are likely to cause the same error (noise) on each wire – often called common-mode noise. If this error is δ then the voltage on wire 1 will become $V_1 + \delta$ and the voltage on wire 2 will become $V_2 + \delta$. The signal is the difference between the two wires, that is $((V_1 + \delta) - (V_2 + \delta)) = (V_1 - V_2)$, which is the same as without the error. For external effects to influence the two wires equally and facilitate cancellation they must have the same impedance – this is referred to as a balanced connection. For this reason, balanced and differential systems are often used in electrically noisy environments or where good signal integrity is required. Common-mode noise may be at a higher frequency than the signal (as in Fig.10), or at the same or lower frequencies.

Differential amplifiers

To amplify a differential signal you need an amplifier with a differential input – called, unsurprisingly, a 'differential amplifier'. The output is often single ended, and the op amp (as seen in Fig.1 and Fig.2) is a well known example of this. Although the op amp itself is a differential amplifier, the commonly used circuits built from it – the inverting amplifier (Fig.2) and non-inverting amplifier – are not differential amplifiers. Amplifiers with both differential inputs and outputs are called fully differential amplifiers.

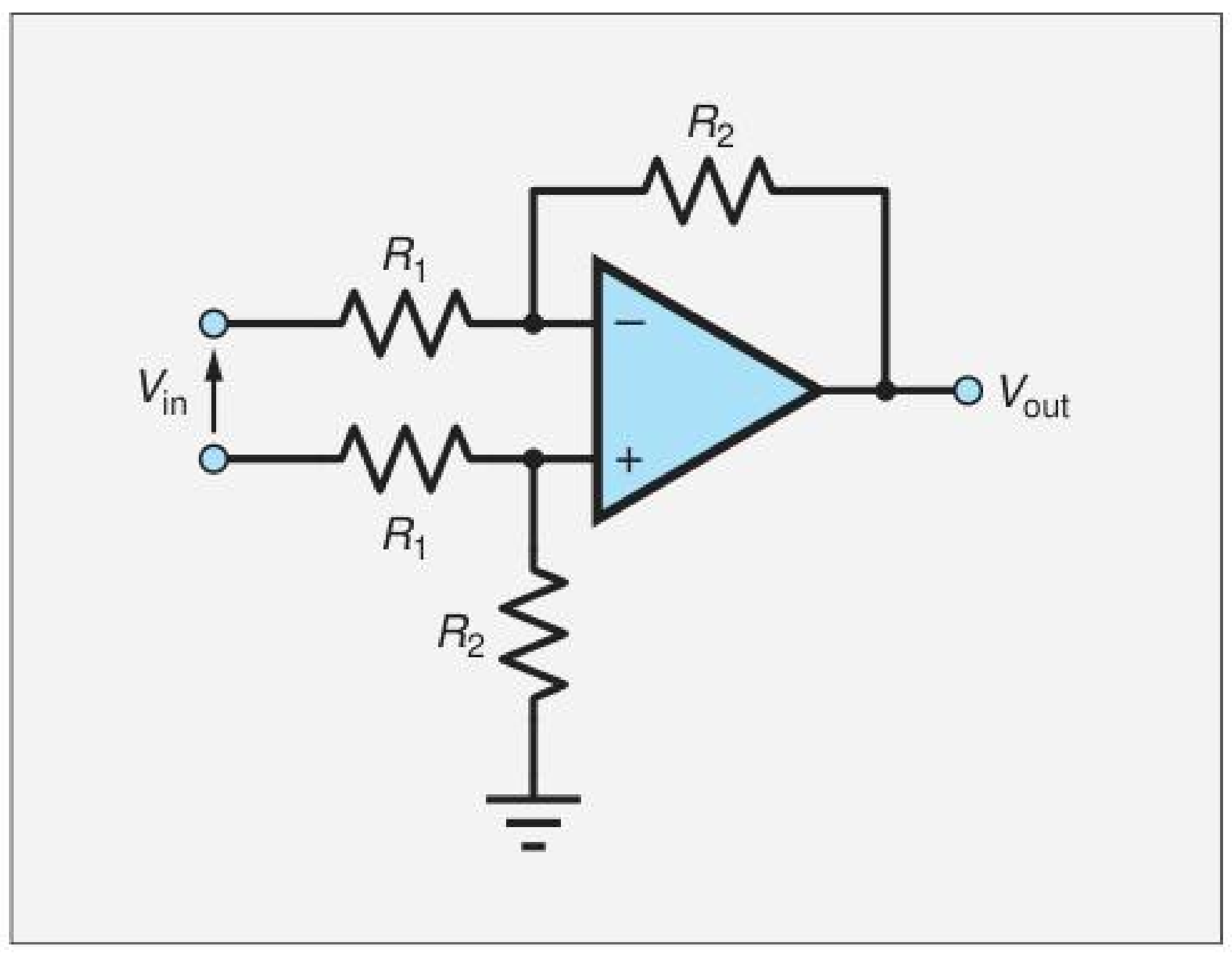


Fig.12. The basic op amp differential amplifier circuit (difference amplifier).

Common-mode rejection

A differential amplifier with inputs V_1 and V_2 amplifies the difference between them ($V_1 - V_2$) by its differential gain A_d. So for an ideal differential amplifier the output is simply $A_d(V_1 - V_2)$. Ideally, a change in common-mode input voltage should not affect the output of a differential amplifier, but in practice it does to some extent. As noted above the common-mode component is the average of the two signals that is $(V_1 + V_2)/2$. This is amplified by the amplifier's common-mode gain (A_{cm}) to produce an unwanted output $A_{cm}(V_1 + V_2)/2$. The smaller the influence of common-mode signals on the amplifier, the better it is. The ability of an amplifier to reject common-mode signals is expressed as the ratio of the differential and common-mode gains; this is the common-mode rejection ration (CMRR), which is often expressed in decibels as follows:

$$CMRR = 20\log_{10}\left|\frac{A_d}{A_{cm}}\right|$$

Typical values for CMRR for op amps are 80 to 100dB (10,000 to 1,000,000).

CMRR is not the only common-mode-related property of amplifiers. The common-mode input range is also important. If a signal has a large common-mode part then this may upset the bias circuits in the amplifier, even if the differential signal is very small. Some differential amplifiers can handle common-mode voltages close to, or even beyond the supply voltages, but others have a much more restricted range. This is something that should always be checked when selecting or designing a differential amplifier.

The concept of CMRR can also be applied to other situations; for example, different strain gauge physical configurations and bridge circuits have different CMRR with respect to influences common to multiple gauge foils, such as temperature changes or tension in the beam arrangement discussed earlier.

Difference amplifier

Although the basic op amp circuits mentioned above are not differential, it is of course possible to build differential amplifiers using op amps – Fig.12 shows a basic difference amplifier op amp circuit. However, it is important to be aware that using an op amp with very good CMRR does not guarantee that the circuit you build with it will also have good CMRR.

The differential gain of the circuit is given by R_2/R_1 – the design assumes that the two R_1 values, and the two R_2 values are exactly the same. Unfortunately, the CMRR of the circuit in Fig.11 depends strongly on this matching of the resistor values. With an ideal op amp and exactly matched resistors the CMRR would be infinite (ideal). With real resistors, however, we get a variation in individual values, which degrades the matching and reduces the CMRR. For example, consider a typical differential amplifier with a gain of 100 in which R_1 is 1kΩ and R_2 is 100kΩ. If the value of one of the R_1 resistors and one of the R_2 resistors is 5% different from the other corresponding resistor then we get a CMRR of only 26dB, even with an ideal op amp. For a 1% mismatch of resistor values we get about 40dB CMRR, for 0.1% about 60dB, but we need 0.01% resistors to get about 80dB (the lower end of typical op amp CMRRs). So we need high precision (and hence relatively expensive) resistors to get even a half-decent performance from this circuit.

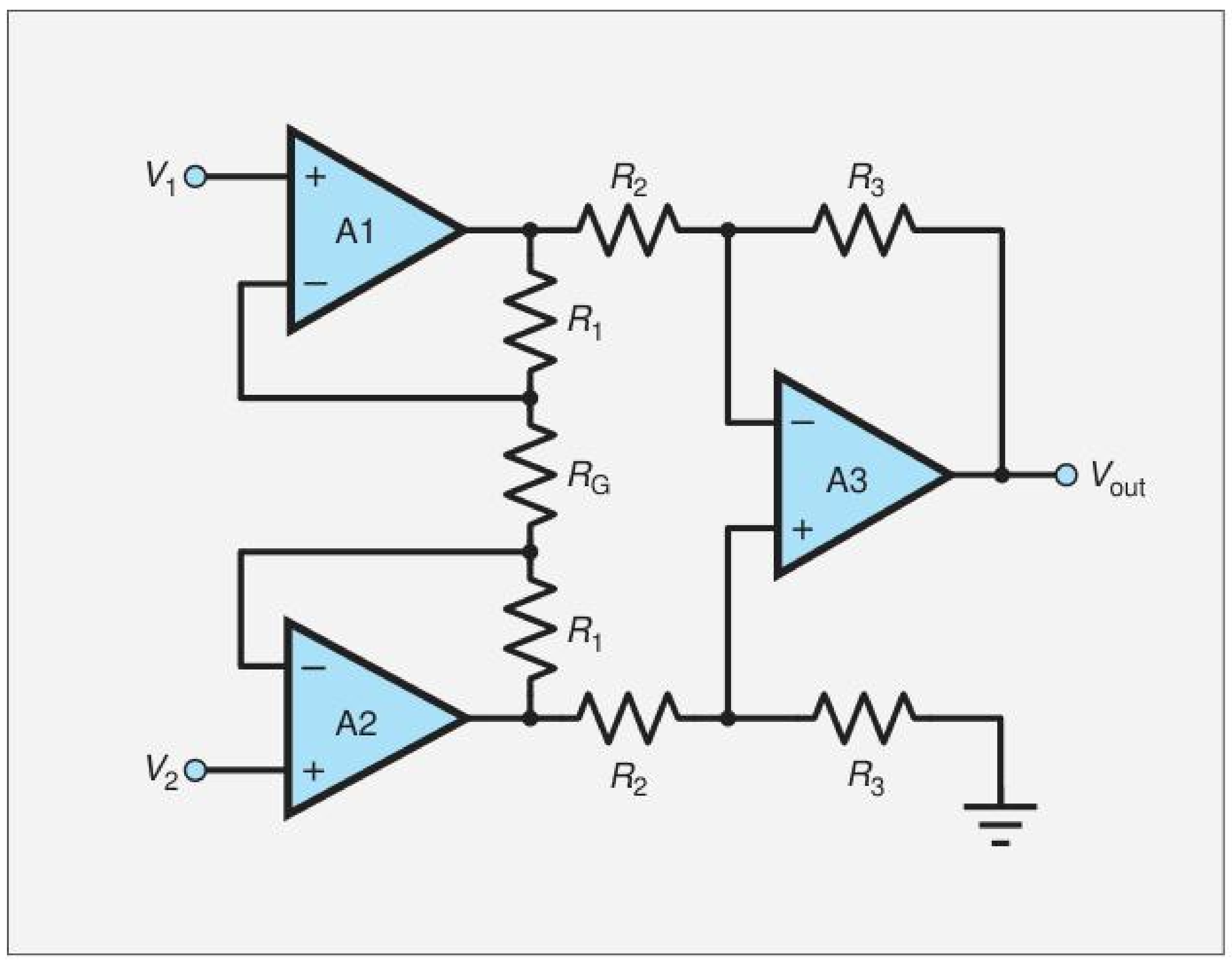


Fig.13. Generic Instrumentation Amplifier.

This problem can be solved by using an integrated version of the whole differential amplifier (referred to as difference amplifiers) where the IC is manufactured with resistors of the required accuracy. These have fixed gains as the resistors are built-in, but some have multiple resistors available to provide different gain configurations. Many such devices are available from well-known manufacturers, such as Analogue Devices and Texas Instruments.

Instrumentation amplifier

Another widely used circuit for amplifying differential signals is the instrumentation amplifier. This is a three-op-amp circuit which is widely used in measurement applications – hence its name. A generic instrumentation amplifier circuit is shown in Fig.13. Op amps A1 and A2 provide very high input impedance by virtue of the direct connection of the signal to the input to the op amp. A1 and A2 also provide gain, set by the values R_G and R_1.

A3 and the R_2 and R_3 resistors form a standard differential amplifier. A1 and A2 buffer the inputs to the differential amplifier (A3) prevent loading of the input source. This buffering could be achieved by two isolated amplifiers, but the cross-connection via R_G results in an output from A1 and A2 which is dependent on both V_1 and V_2. This enhances the differential gain of the whole circuit and improves CMRR.

You can build an instrumentation amplifier with discrete components, but its performance may not be very good, particularly in terms of common-mode rejection. Like the difference amplifier circuit, it requires very close matching of resistors (eg, the two R_3 resistors in Fig.13 must have exactly the same value). Characteristics of A1 and A2 should also be closely matched. Just as with the difference amplifier, this matching can be achieved when the whole circuit is built as an IC, and again, many examples are available from the semiconductor companies.

Fully differential

Finally, there are fully differential amplifiers, with differential outputs as well as inputs. In circuits where errors need to be kept to a minimum (such as measurement systems) keeping a signal in differential form throughout the processing can provide a big advantage. This tends to cancel a significant amount of unwanted effects; however, the circuits are, of course, more complex. Given that most measurement systems deliver a signal from a sensor to a data converter it is worth noting that many ADCs have differential inputs and are designed to work with compatible fully differential amplifiers – this option was discussed in the forum response to Scott's post about his strain gauge amplifier.

Simulation files

Most, but not every month, LTSpice is used to support descriptions and analysis in *Circuit Surgery*. The examples and files are available for download from the *PE* website.

PIC n' Mix

Mike Hibbett's column for PIC project enlightenment and related topics

Small, cheap and powerful – Part 4

After a hectic summer recess, we now return to the subject of doing interesting things with very small, very cheap and very limited microcontrollers. We continue from where we left off last time – using a PIC10F202 processor, an eight-pin microcontroller with 512 words of Flash and 32 bytes of RAM. Last time, we were reading a switch and controlling an LED. This month, we explore the important topic of connecting to other, more complex devices.

Serial interface choices

There are many different ways in which digital ICs connect to each other, but communication between 'smart' devices generally means either a serial or parallel interface. A parallel interface (typically) uses eight wires to send each data byte and requires several other signals to provide handshaking between the two chips. Parallel interfaces can be very, very fast, but are clearly not an option for us on an eight-pin processor package. Serial interfaces, on the other hand, use just a few signals – sometimes as few as one – to send data. There are four common serial interfaces available; let's take a brief look at each of them.

Serial Peripheral Interface bus

SPI requires four wires for bi-directional communication (see Fig.1), but sometimes, when you do not need a return signal from the connected device, you can get away with three wires. The 'master' controls the rate of communication, which in our case is the processor. Exchange of data is very strictly controlled – the master sends the required data bytes, and then 'clocks' the expected data back in. This simplifies operation in that the exact data rate is not terribly important, as data bits pass between the two devices on the transition of the clock signal generated by the master. Data rates can be what you want, below the maximum rate the connected device can handle, which is often in excess of 10MHz.

Multiple devices can be connected to the same SPI bus, but each device will require its own GPIO pin to control its Slave Select (SS) line. As the number of devices connected goes up, so does the required pin count. Data rates can, however, be quite high, above ten million bits per second, which does makes it attractive for high data-rate requirements.

I²C bus

The Inter-Integrated Circuit bus, usually just called 'I²C', differs from SPI in that it requires only two signal lines for communication, named 'SCL' (the clock signal generated by the master device) and 'SDA' (the data line shared by the master and slave device.) See Fig.2.

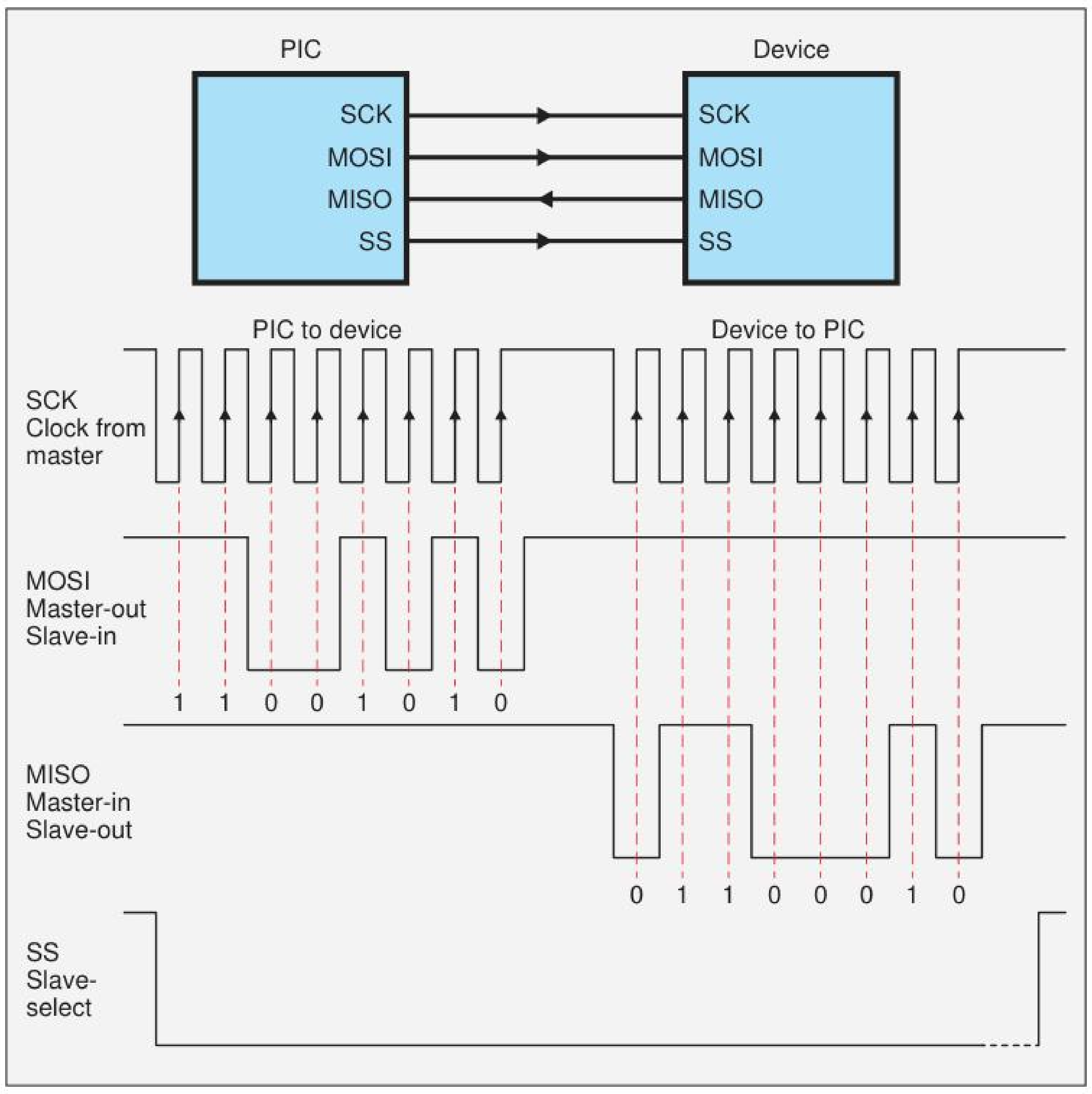


Fig.1. SPI bus operation – a four-wire system.

Adding more devices to the bus does not require extra wires; each device has its own 8-bit address and will only respond to messages addressed to it. I²C is a slower interface than SPI and slightly more complicated to implement, but its 2-wire interface makes it very efficient on processor pin allocation, and so will be ideal for us – if we can make the code fit. I²C was invented in the early 1980s by Philips, remains a very popular bus interface and there is a huge range of ICs and devices that make use of it. I²C differs from the other three protocols mentioned here in that data is transmitted most-significant bit first, rather than least significant.

UART

The UART protocol (Universal Asynchronous Receive Transmit – what a mouthful) shown in Fig. 3 is a very old protocol that has been around since before the dawn of microcontrollers. The older generation will know it as RS232, which is actually the specification for the physical interface over which the signals are transported. The protocol is relatively simple but precise timing is required because each end of the data link expects the rate at which signals are exchanged to be a

PIC n' Mix files

The programming files discussed in this article are available for download from the *PE* website.

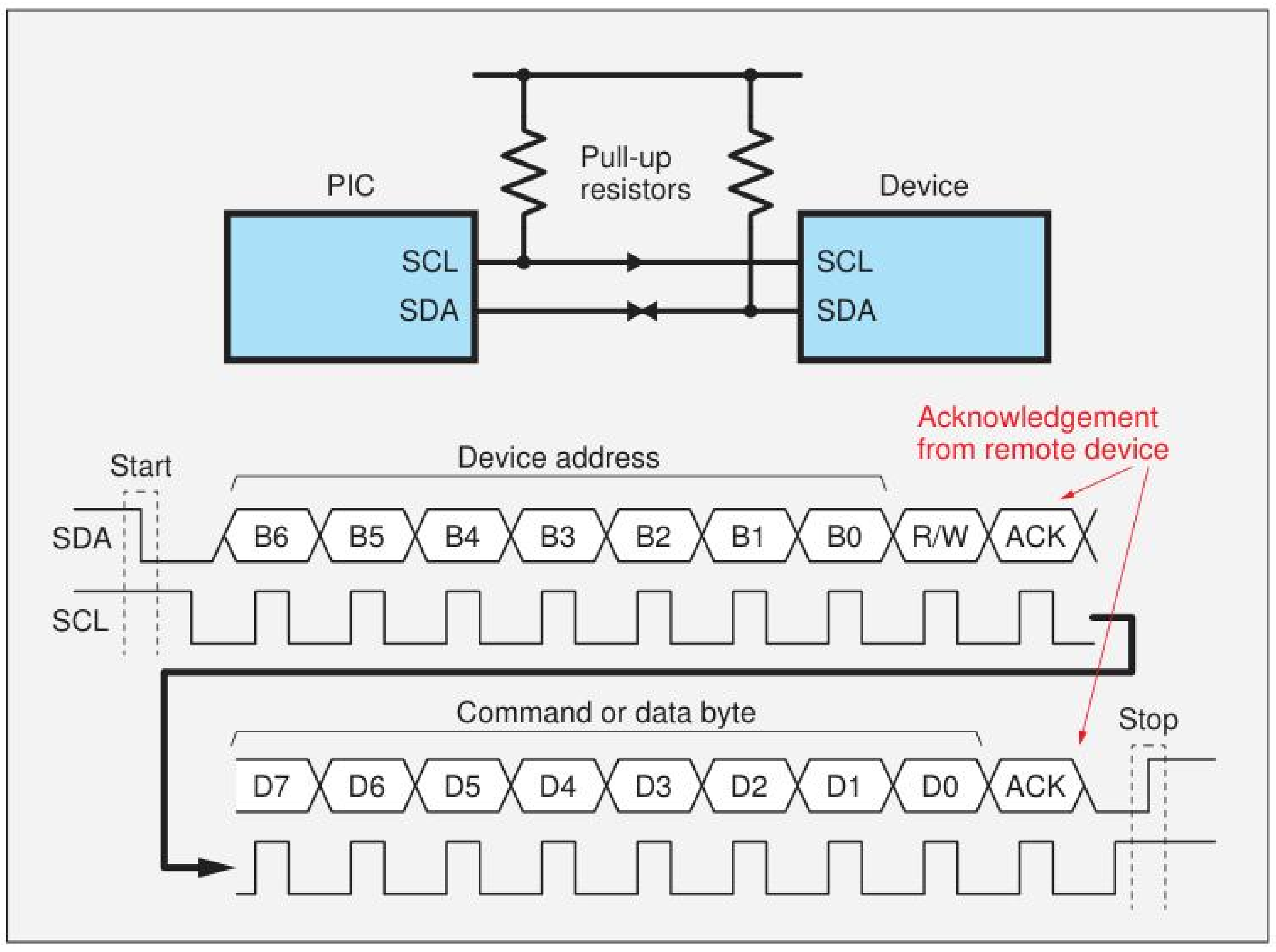

PIC
Pull-up resistors
Device
SCL
SDA
SCL
SDA
Acknowledgement from remote device
Start
Device address
SDA
B6
B5
B4
B3
B2
B1
B0
R/W
ACK
SCL
Command or data byte
Stop
D7
D6
D5
D4
D3
D2
D1
D0
ACK

Fig.2. I^{2}C bus operation is implemented with two signal lines. Note that it uses nine, not eight bits for data transfer.

specific rate, known as the 'baud rate'. Each device must transmit data at a rate within 2% of that specified to avoid data corruption or data loss. Baud rates can, however, be selected from a wide range of speeds, from 1200 bits per second up to two million bits per second, providing each device supports these rates. Implementation of the protocol in software is reasonably straightforward, but the physical interface is purely point to point, so it is impossible (without some hacks) to implement a solution with multiple devices on the same UART interface.

1-Wire

The 1-Wire protocol is an unusual protocol, based, as it states, on a single wire used to communicate bi-directionally between two devices. Like most marketing claims this is of course misleading – a ground signal shared between the two devices is also required, so it's actually two. Despite this, the 1-Wire protocol is quite popular in some specific areas, namely access control and temperature sensing. You can see from Fig.4 that it is quite a complicated protocol with very specific timing requirements. In theory, we could implement this on the PIC10F202, but due to the limited memory it is unlikely that we would be able to implement any other communication protocols.

This month's project

Having highlighted the options available to us, let's choose a design that will stretch our microcontroller to its limits while creating an interesting and potentially useful circuit.

Having a display of some kind attached to our circuit would of course be nice. The most common display used by hobbyists is the cheap 2 × 16 character LCD, but these typically have parallel interfaces, which we cannot use as we have insufficient GPIO pins available. However, there are LCDs that include I^{2}C or UART interfaces, still reasonably cheap and available on eBay and Amazon. Some run at 3.3V, others at 5V – see Fig.5.

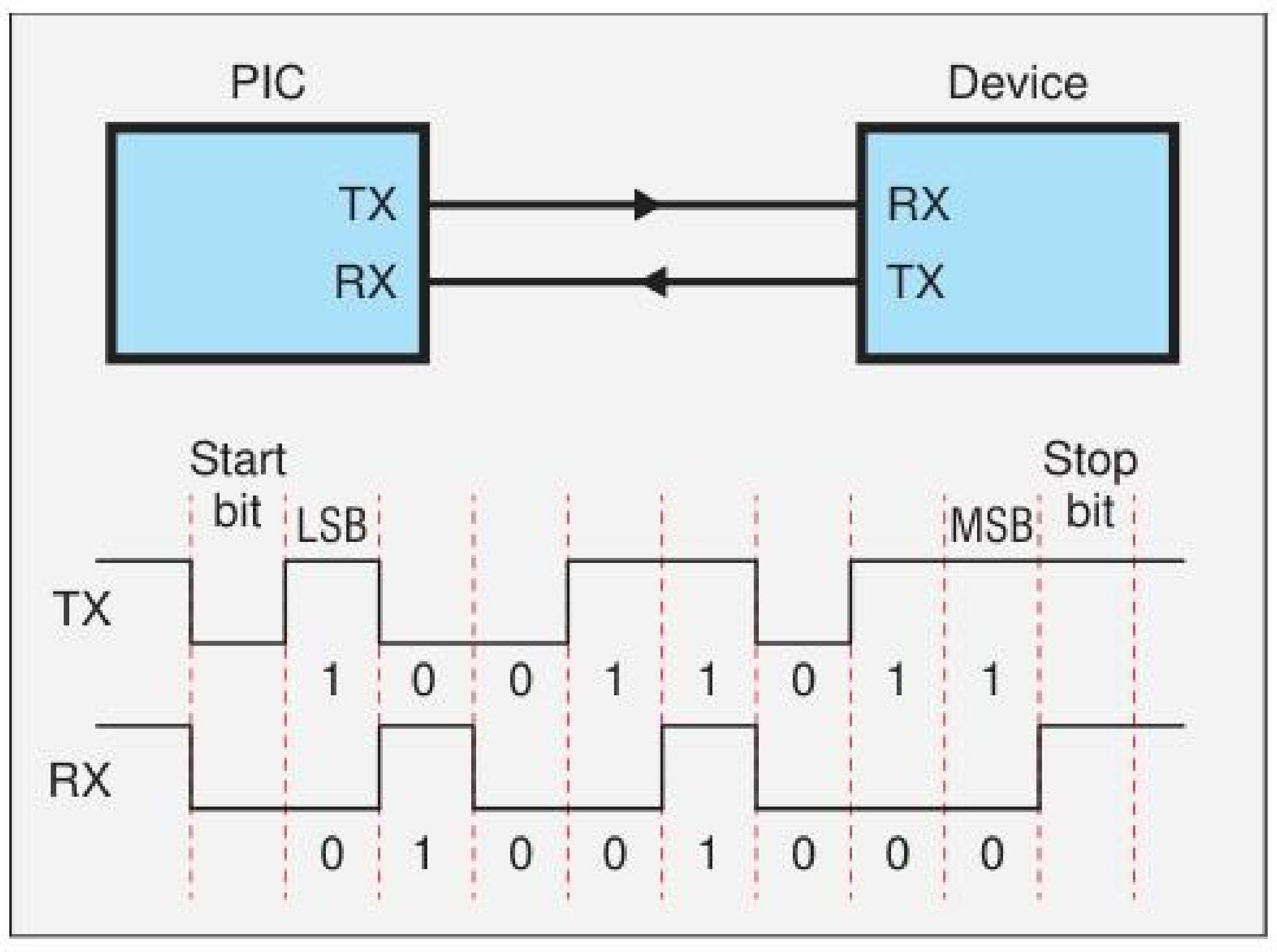

PIC
Device
TX
RX
RX
TX
Start bit
LSB
MSB
Stop bit
TX
1
0
0
1
1
0
1
1
RX
0
1
0
0
1
0
0
0

Fig.3. UART communication. Signal format on RX and TX pins are identical.

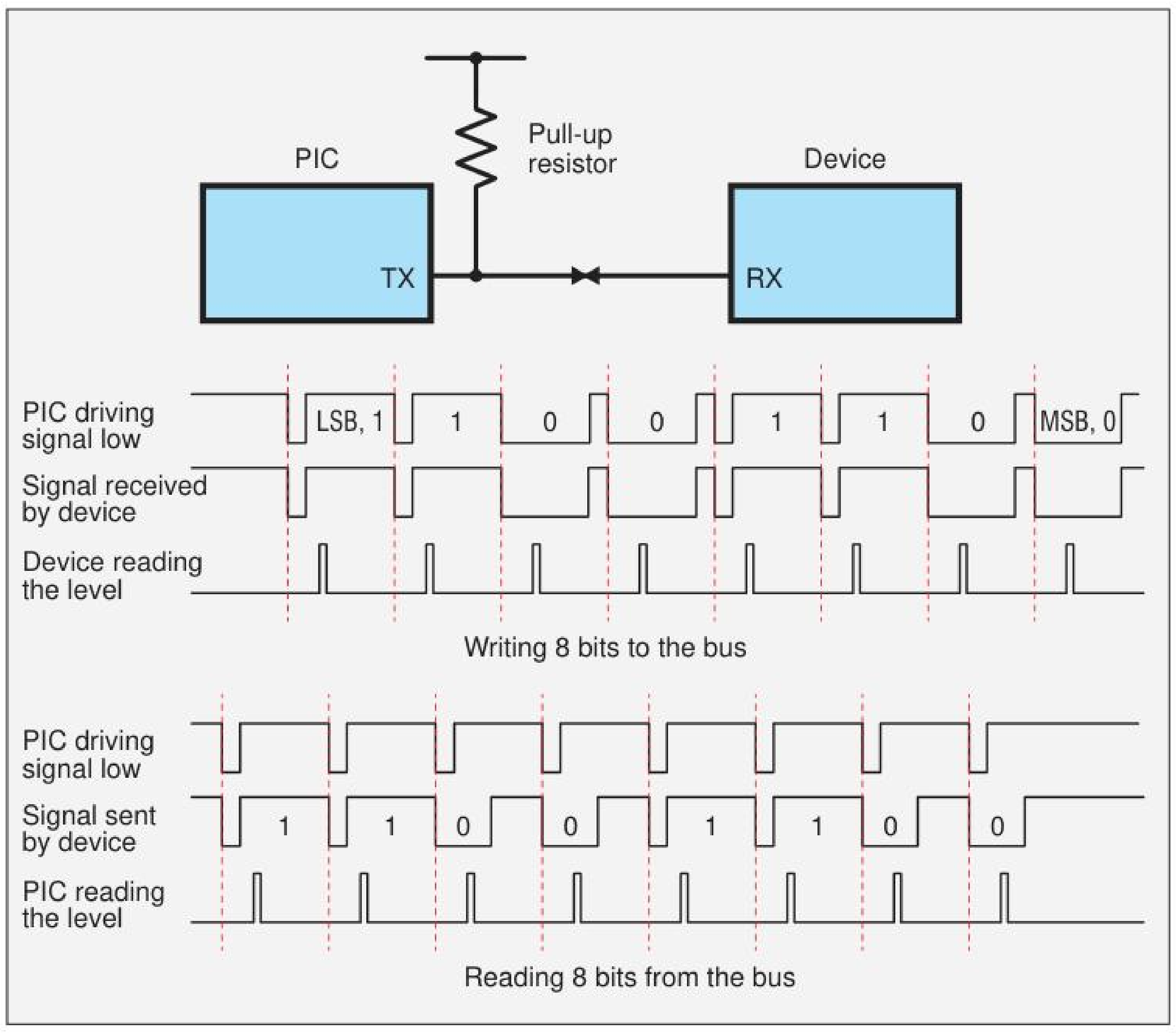

PIC
Pull-up resistor
Device
TX
RX
PIC driving signal low
LSB, 1
1
0
0
1
1
0
MSB, 0
Signal received by device
Device reading the level
Writing 8 bits to the bus
PIC driving signal low
Signal sent by device
1
1
0
0
1
1
0
0
PIC reading the level
Reading 8 bits from the bus

Fig.4. 1-Wire serial communication operation. It is quite a complicated protocol, and of course it's not really '1-Wire' – a second ground wire is always needed.

We have chosen to go with a 5V I^{2}C based LCD module. The logic behind this decision is that if we implement the I^{2}C driver to communicate with the display, the same code can be reused when we attach additional devices such as temperature, humidity or acceleration sensors. There are even magnetic compass modules available, so taking this route – using I^{2}C rather than UART, SPI or 1-Wire – should give us the greatest flexibility for the future.

Let's dive in. The schematic for this month's circuit is shown in Fig.6. As you can see, we are using a 5V I^{2}C 2 × 16 line LCD display, this one from Grove because it is readily available. Besides having an I^{2}C interface for control, it has a rather cool RGB backlight that is under full user control. Cheaper modules are available on eBay; we rather like the 'LCD RGB KEYPAD For RPi', which at just £4 includes a small keypad. We would have loved to use one but these kind of products are often only available for sale for a short time (there appear to be just five left on eBay) so do not make good article material. Hooking our processor up to one would be simple however, so do feel free to experiment

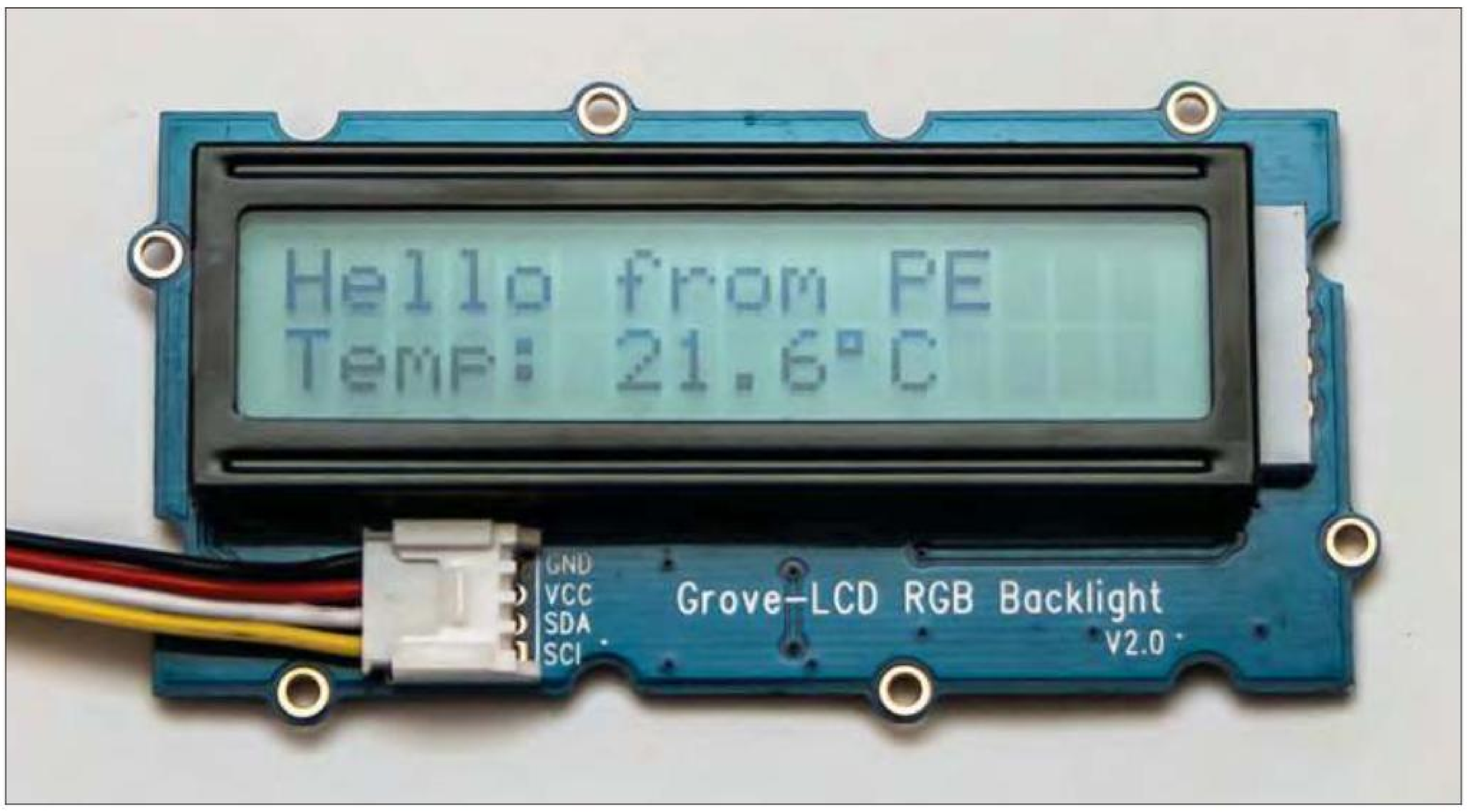

Hello from PE
Temp: 21.6°C
GND
VCC
SDA
SCL
Grove-LCD RGB Backlight
V2.0

Fig.5. 2 × 16 I^{2}C LCD display, of the type used in this month's project.

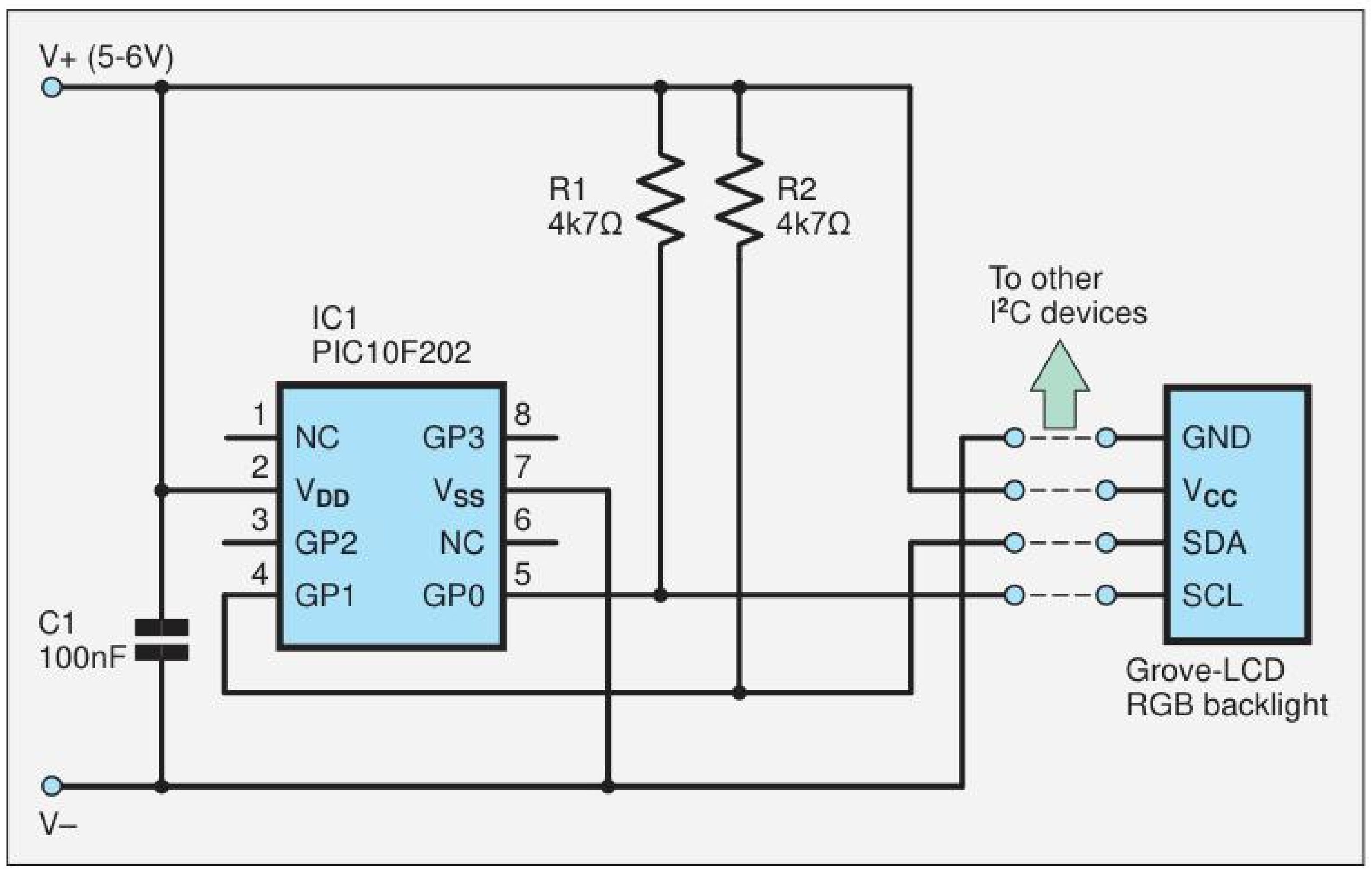


Fig.6. Using our minimalist PIC10F202 microcontroller to drive an I^{2}C LCD display.

with personal projects on these short-lived bargains. If you are only planning to build one or two devices, they are well worth it. You can find the Grove LCD module on eBay for around £11 from a number of suppliers – for example, Digi-Key part 1597-1336-ND.

I^{2}C driver

Moving back to software development, we'll now look at what I^{2}C commands need to be given to the Grove LCD to set it up. The driver IC on the LCD is still based on the HD77480 (a decades-old device) so the initialisation sequence is the same as for the parallel interface LCDs – the only difference is that the commands and data are sent using the I^{2}C bus rather than a four or eight-wire parallel bus.

As well as a datasheet, there are also software drivers available for the Grove LCD written for the Arduino development environment that we can use to understand what control messages need to be sent to the display to initialise it. First though, we need to write the low-level I^{2}C driver software, and get it to fit within our microcontroller with enough memory space left over for an actual application.

Before writing any code we start by organising our project into some new files. So far in this series we have placed all our code in the **main.c** text file. Adding in the I^{2}C driver code to this file will increase the size and make finding particular sequences of code difficult. We will create a new file – **i2c.c** – which will contain just the I^{2}C driver code. To gain access to the functions within our application program file **main.c**, we will create a file called **i2c.h**, which will contain the I^{2}C function declarations. We include a reference to this file in **main.c**, near the top, like this:

```
#include "i2c.h"
```

This is standard C language practice. If you are new to the C programming language and get confused about different 'source files', 'header files' and such like, we have found the best way to understand this is to understand how the C compiler operates. The complier program does not care about header files or source files, it just works on the text files you ask it to work on. It cares nothing about .c and .h file extensions; this is just a convention that programmers have decided to use. You could call a header file **mycode.headerfile**; the compiler does not care, you just have to specify that name. Having said that, no one else will understand your filenames, so if you intend to share your code with others, stick with the conventional naming.

Once the I^{2}C functions have been written in the file **i2.c**, all of our 'application-specific' code will live in the file **main.c**. This will make it far easier to reuse the I^{2}C functions in a new project; for future projects you just copy the files **i2c.c** and **i2c.h** and start using them – no need to edit our application-specific code.

Now let's dig a bit deeper into the I^{2}C driver software itself. Referring back to Fig.2, you can see the I^{2}C operation consists of three distinct phases:

- Start condition
- 9-bit transfer(s)
- Stop condition

A start condition is signalled to all devices on the bus by the SDA pin going from high to low while the SCL line is high. The stop condition is indicated by the SDA line going from low to high while the SCL line is high.

The nine bits of data are transferred, one bit at a time, when the SCL pin goes from low to high. The SDA pin is set to the correct level for each data bit while the SCL signal is low, which avoids triggering a start or stop condition. When data is being transmitted by the master – our processor – we simply set the level of the SDA pin before toggling the SCL pin. When we are reading from the slave – a temperature sensor for example – We set the SDA pin to an input and read its level just after toggling the SCL pin. It's a very nicely thought out communication technique.

There are no tight timing requirements on the rate at which data is transferred; everything is synchronised to the transitions on the SCL pin. Specific devices will have their own maximum clock rate limitations, but the specification limits these to three maximum rates – 100kHz, 400kHz or 1MHz. However, they do not care whether the transfer occurs at a variable speed lower than the limit. This makes the design of our code easier, as we do not require accurate bit timings, as we do with UART or 1-Wire communication.

You may have spotted an odd point about the data transfer – it's nine bits in length, not eight. The ninth and final data bit in the transmission is the ACK bit (acknowledgement.) This is not a bit transmitted from the microcontroller, but a response from the device back to the microcontroller it is talking too, and vice-versa when the microcontroller is receiving data from the external device. To receive the ACK bit, at the end of transmitting eight data bits the microcontroller will turn its SDA pin to an input and look for the SDA pin to be pulled low during the SCL transition. If the pin is pulled low, this signals the remote device is ready for another data byte. If not, it signals an error, or a request to stop further communication.

Code implementation

Based on the signal sequences shown in Fig.2 we have decided to implement this driver as four distinct functions within the file **i2c.c**:

```
void i2c_start(void)
void i2c_stop(void)
unsigned char i2c_tx(unsigned char tx_val)
unsigned char i2c_rx(unsigned char ACK)
```

The `i2c_tx` function takes the value you want to send on the bus, and returns the value of the ACK bit. The `i2c_rx` function returns the value sent by the external device. It takes a parameter that indicates what signal should be asserted during the final ninth data bit, the ACK bit. Remember, the ACK bit is sent by the remote device when we are sending data to it; when we are receiving data then we are the remote device, and typically use the ACK bit to indicate that no further data is required.

The source code and build files for this project are available on the November 2019 page of the *PE* website, but let's take a closer look at one of the I²C functions, `i2c_start()`, to understand how it works:

```
void i2c_start(void)
{
SDA_HIGH;
__nop();
SCL_HIGH;
__nop();
SDA_LOW;
__nop();
SCL_LOW;
__nop();
}
```

Note that statements such as `SDA_HIGH` and similar text in capitals are macros – these get converted by the compiler (actually by the compiler pre-processor, but that doesn't really matter) into short sequences of code that manipulate the SDA and SCL pins directly. We have defined these macros in our **i2c.h** header file. Using macros is just a form of shorthand – it makes our code more readable and helps reduce errors when writing software. The `__nop()` function translates directly into the assembly instruction `NOP`. These `__nop()` calls are made to allow a very small delay when toggling pins to avoid clocking the remote device too fast. A single `__nop()` sequence is sufficient due to our processors low clock speed.

Now let's look at the LCD initialisation routine. This function lives in the **i2c.c** source file, as it is a generic start-up sequence for an I²C LCD module. It is shown in Listing 1, and demonstrates how the I²C functions are used. Each command to the display is a single transaction consisting of a start condition, three bytes and a stop condition. The actual sequence of commands was taken from the Open-Source Arduino library, although they are also clear in the datasheet.

One of the nice features of the C programming language is that any functions in you application source files that are not used will not be copied into Flash memory. This allows you to construct template projects with all your libraries present, while consuming no processor memory unless you actually use them.

Performance

This is a very minimal implementation of an I²C bus interface and lacks low-level error detection, handling or recovery. This means that robustness to error conditions such as noise on the data signals should be handled within your program, or simply accepted as likely causes of program lock ups if problems occur. For hobbyist use, these limitations are excusable as you can simply power cycle the device if it stops functioning. For a business-critical application however, you would choose a more powerful device with better error-handling capabilities.

The I²C driver takes up 259 words, just 51% of our code space. This leaves a lot of space for communicating with another device, and writing information to the LCD.

The circuit, as shown, draws 10mA at 5V with the backlight on, just 1mA with the backlight off. This would give about four months of continuous operation from four AA batteries.

Listing 1: Source code for I²C operation

```
void LCD_init(void)
{
  delay_50ms(5);

  // Send first command
  i2c_start();
  i2c_tx(LCD_ADDRESS);
  i2c_tx(0x80);
  i2c_tx(LCD_FUNCTIONSET | LCD_2LINE);
  i2c_stop();

  delay_50ms(1);

  // resend first command in case of
  // a startup failure

  i2c_start();
  i2c_tx(LCD_ADDRESS);
  i2c_tx(0x80);
  i2c_tx(LCD_FUNCTIONSET | LCD_2LINE);
  i2c_stop();

  delay_50ms(1);

  // resend first command in case of
  // a startup failure
  i2c_start();
  i2c_tx(LCD_ADDRESS);
  i2c_tx(0x80);
  i2c_tx(LCD_FUNCTIONSET | LCD_2LINE);
  i2c_stop();

  // Set display mode
  i2c_start();
  i2c_tx(LCD_ADDRESS);
  i2c_tx(0x80);
  i2c_tx(LCD_DISPLAYCONTROL | LCD_DISPLAYON
  | LCD_CURSOROFF | LCD_BLINKOFF);
  i2c_stop();

  // Clear display, set cursor to top left
  // position
  i2c_start();
  i2c_tx(LCD_ADDRESS);
  i2c_tx(0x80);
  i2c_tx(LCD_CLEARDISPLAY);
  i2c_stop();

  delay_50ms(1);
}
```

Next time

In the next article, we will look at adding an I²C digital temperature and humidity IC to our circuit, and see if we have enough code space available to create a functional temperature monitor. If we do, we will progress to making our first tiny PCB. A whole host of interesting low-cost sensors are available that use the I²C interface, and can work with our microcontroller. For example, the DHT12 sensor provides humidity and temperature measurements and is available cheaply on eBay and Amazon. Sensors can be purchased in a variety of physical sizes, from easy-to-use wire-ended packages to tiny ICs that can be integrated into a small product. We will experiment with the wire-ended DHT12 in our next instalment.

AUDIO OUT

By Jake Rothman

Speaker nuts and bolts – Part 1

I'm going to have a short break from some of the technicalities of the BBC LS/35A design – one of our main suppliers, Jerry Bloomfield at Falcon, is so inundated with orders for LS3/5As that we are going to have to hold off for a little while. However, the time will not be wasted; we're going to look at some general speaker construction basics, such as connectors, leads and other speaker 'nuts and bolts' tips. All of this is directly relevant to building LS3/5As, and we will use many of these ideas when we build them.

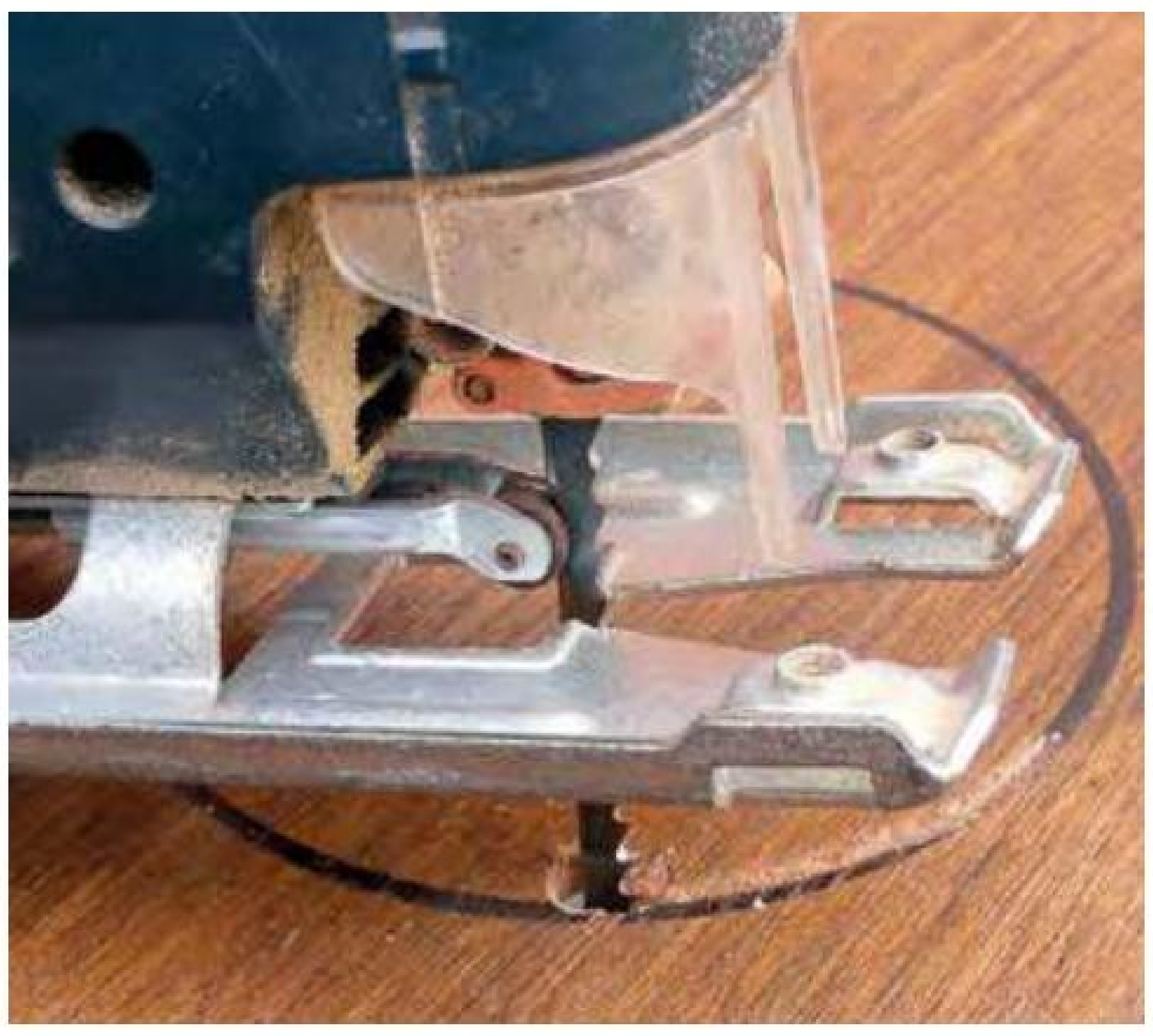


Fig.1. Cutting a speaker hole the crude way using a jigsaw. Note the drilled starter hole *inside* the circle.

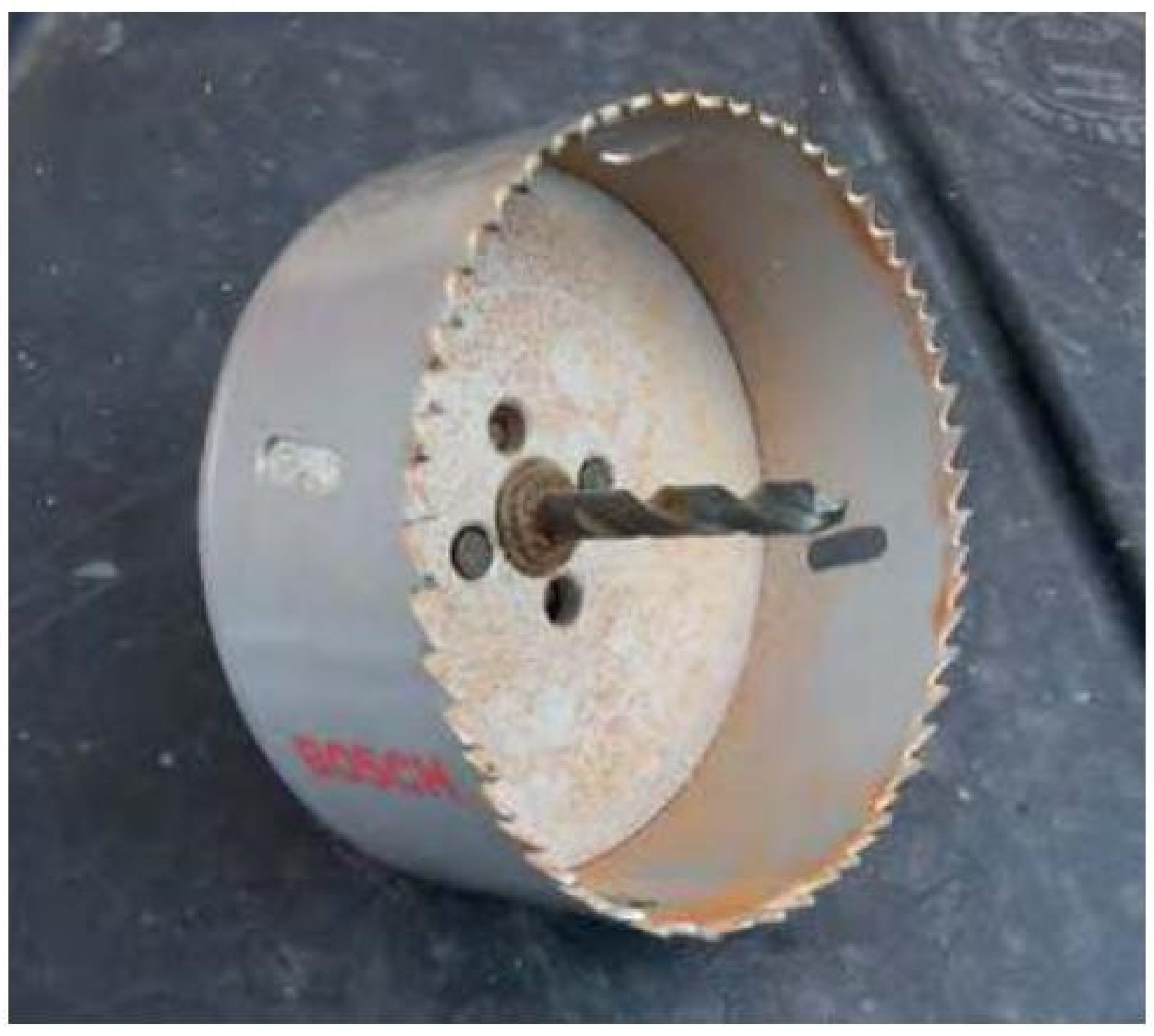


Fig.2. A proper toothed hole cutter gives a more accurate hole. You must lift the cutter often to clear saw dust/shavings.

Drive-unit mounting

A lot of this is basic woodwork. I can remember my school woodwork teacher (in the days before excessive health and safety 'killed' practical work) saying in his gruff Northern accent, 'measure twice, cut once', 'cut on the waste side of the line you wally' and, of course, in my case an exasperated, 'not another blinking speaker box', or words to that effect.

We're making speaker cabinets, but that doesn't mean you have to be a skilled cabinetmaker. Most of the parts we need are accurately cut rectangular pieces of plywood – six per speaker. If you are confident with this kind of woodwork then do it yourself. If not, then a good wood merchant will be able to cut them for you. Remember, we want:

- Straight edges
- Precise dimensions
- Accurate right angles.

Probably the trickiest task is cutting the mounting holes. For just a pair of speakers, a thin pad saw or an electric jig saw works well. Draw a circle in the right place and drill a hole inside its edge to accommodate the saw blade (see Fig.1). It's good practice to pull the blade back frequently to be able to turn it as you go round, you don't want a polygon-shaped hole. For repetitive production work, I use a toothed hole cutter in a pillar drill (see Fig.2). These are quite expensive, one is needed for each size and you need to buy an auger to couple it to the drill chuck.

Whenever you use power tools do wear ear protectors and glasses. I was shocked to hear that an audiometric survey of 229 attendees at an Audio Engineering Society (AES) convention in 1986 found that audio engineers had worse hearing than average, due to noise exposure.

Box assembly

Speaker boxes must be air tight. A loudspeaker cabinet is a pressure vessel – imagine the havoc a small hole would do to a trumpet or car exhaust system. You need to use more glue than normal for the woodwork to seal all the gaps. It should squeeze out as the woodwork is screwed together. I use PVA woodwork glue, such as Evostik Resin W. The excess is simply wiped off with a damp cloth. It then dries clear so you can't see it.

You must let the glue dry thoroughly before mounting the drivers or you will end up with rusty magnets as the glue's water evaporates. If a solvent-based glue, such as impact adhesive is used, the vapours given off as it cures could deform plastic-based diaphragms. This also applies to the mounting of damping pads – wait for the adhesive to properly dry.

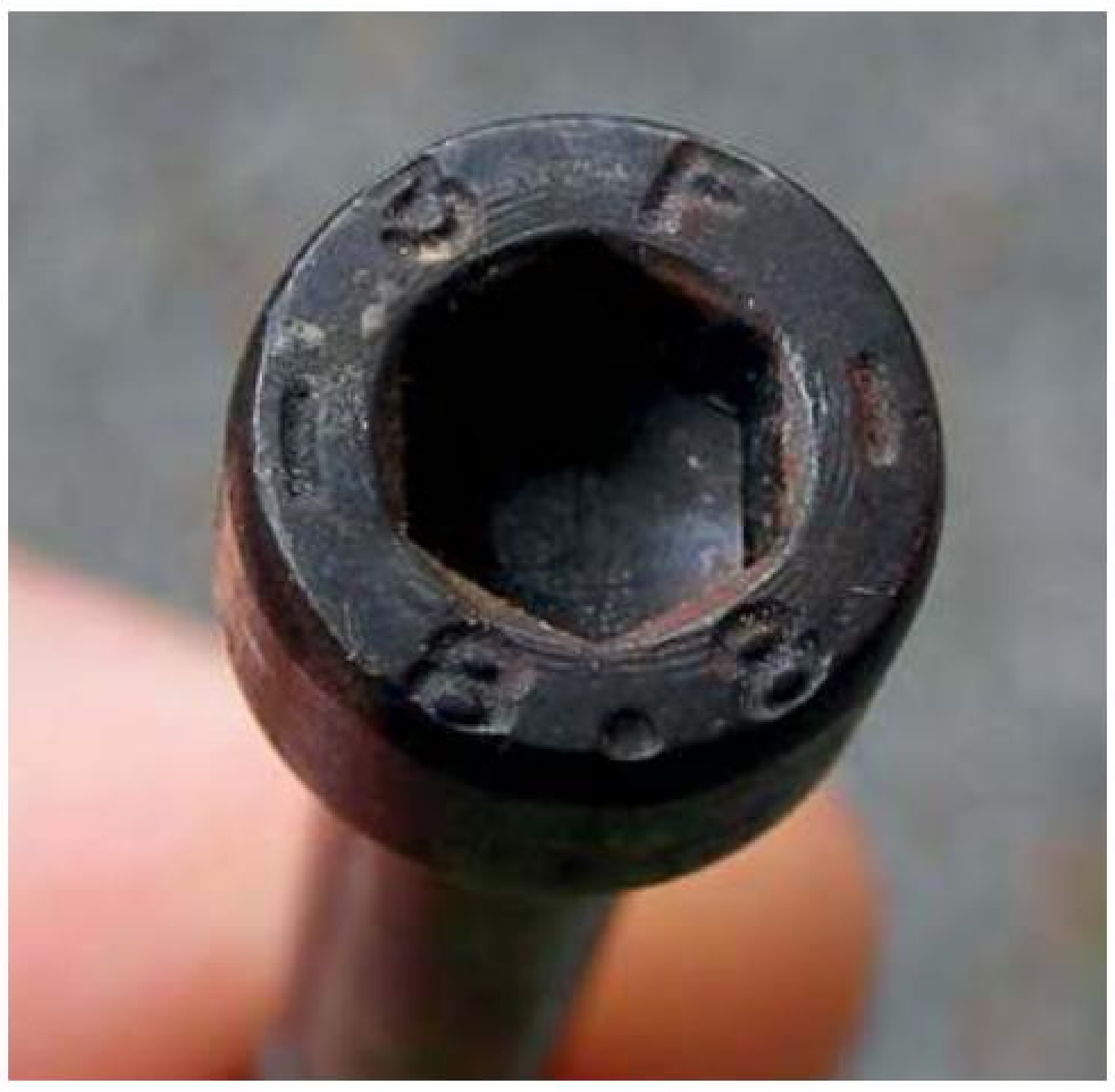


Fig.3. Hex socket screws tightened by Allen keys greatly reduce the chance of cone damage from a slipped screwdriver.

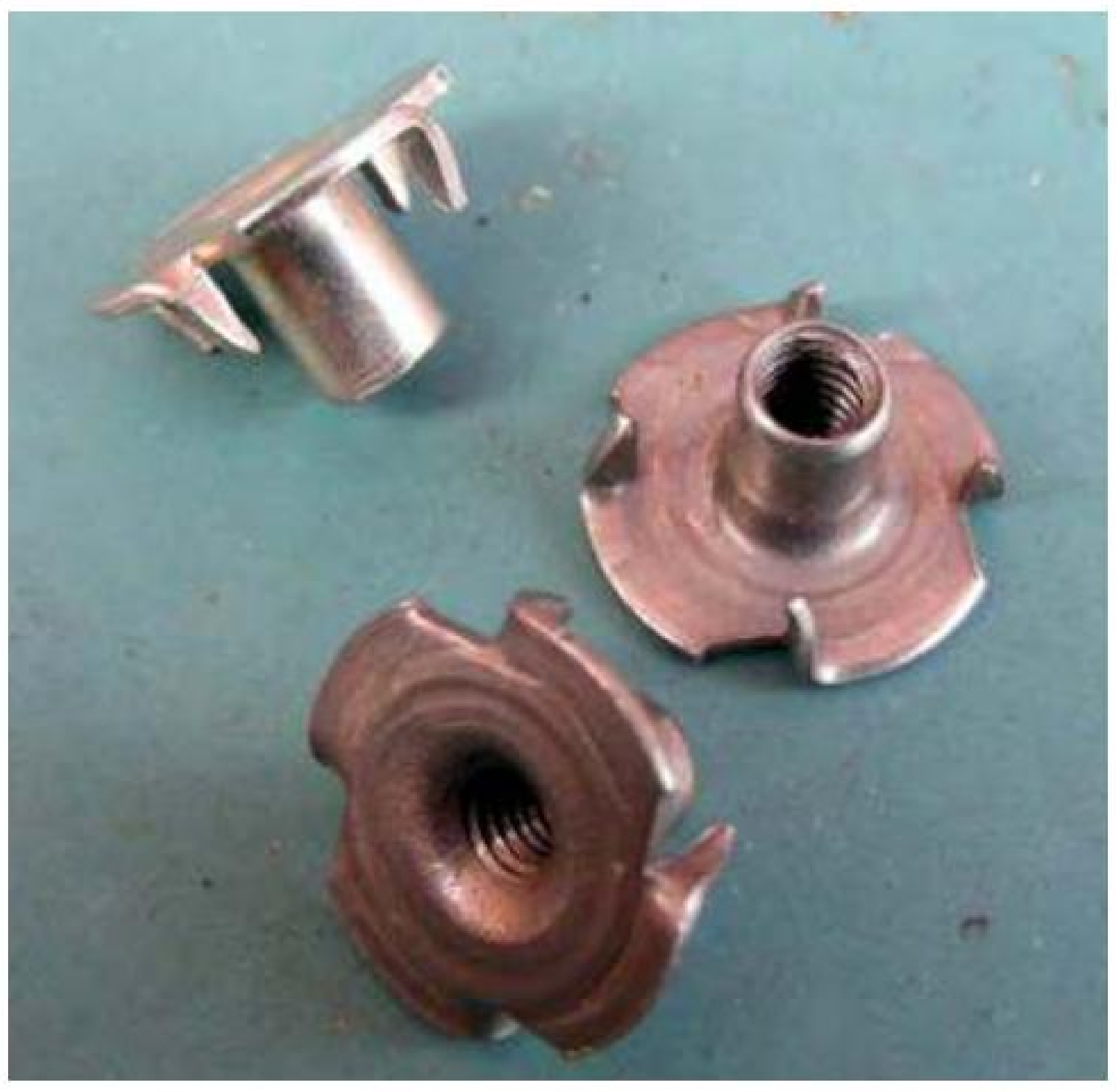


Fig.4. T-nuts provide a metal threaded insert that can be fixed into the wood of the baffle.

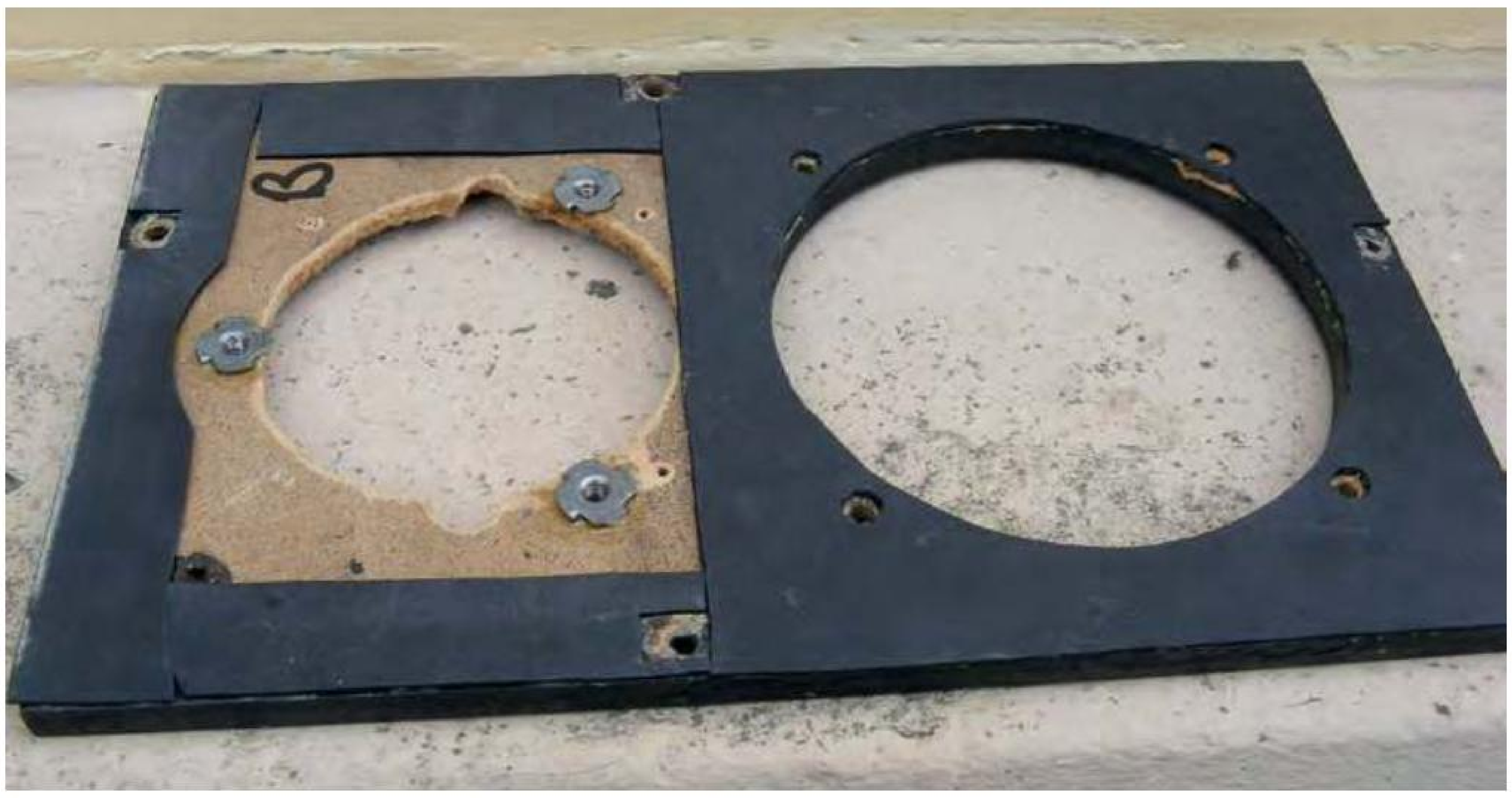


Fig.5. T-nuts put in place for a T27 tweeter on the rear of the baffle (inside the cabinet). SInce the T27 is front mounted, it neatly covers the 'rough' cutout.

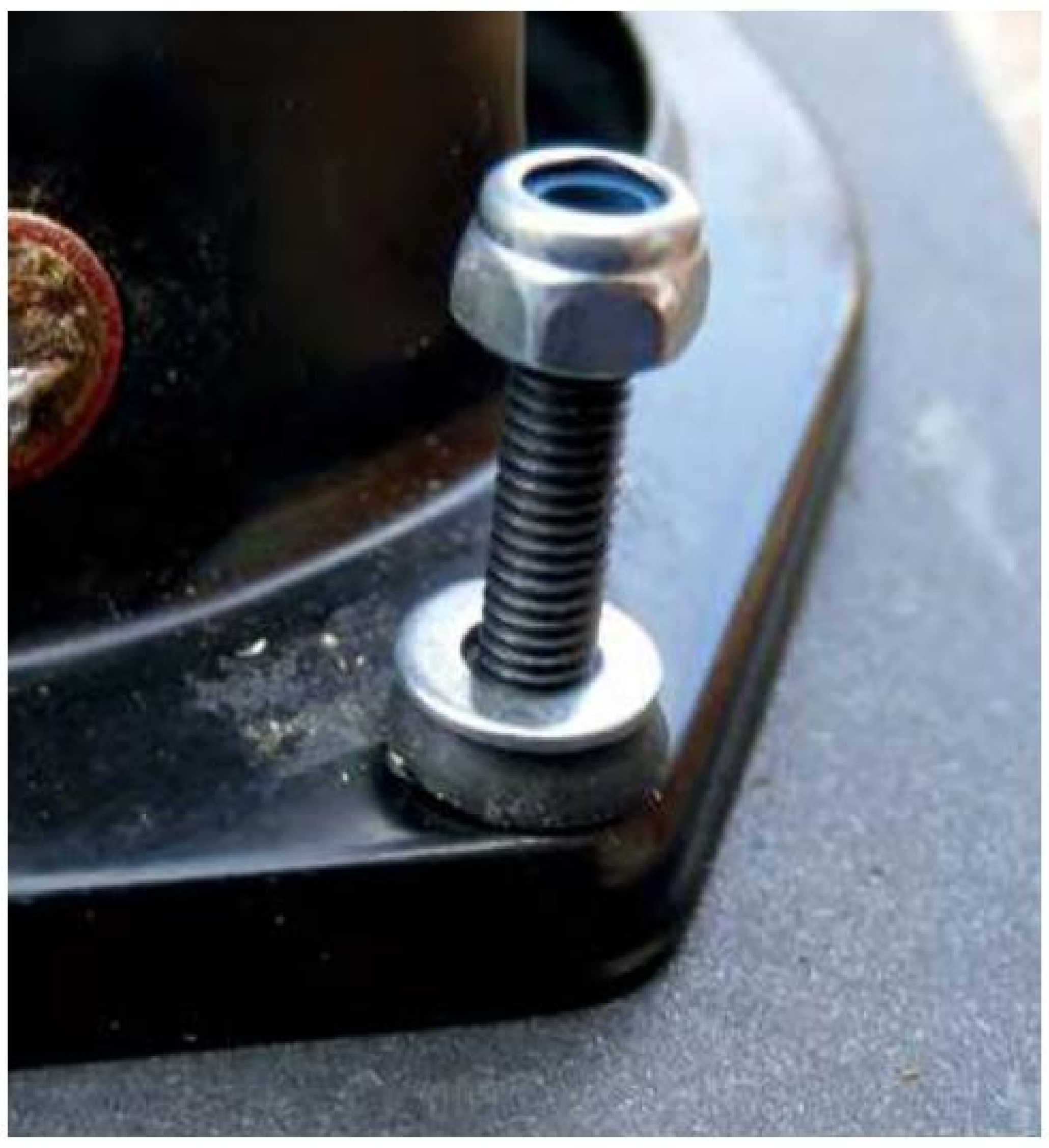


Fig.6. Nyloc nuts avoid the need for locking washers and don't work loose.

Speaker builders use the 'push test' to check if the enclosure is airtight – the bass cone is slowly depressed to squeeze the air out of the box and then released. It should return slowly over a few seconds. If it comes back quickly, then that suggests there is a leak and the air is rushing back in. Also, you can listen for hissing sounds if the speaker is driven with a very low frequency of a few Hertz. It is almost impossible to make a speaker completely air tight – if you did, it would act as a barometer! For a reflex-design cabinet, pressing the palm of your hand across the port can usually seal it while doing the push test.

Front mounting of drive units is best because it hides any minor inaccuracies in the holes and it is relatively easy to make airtight. Most drivers are supplied with foam rubber gaskets, which seal any gaps. I often use black Plasticine to fill edges around flanges; it can be neatly cut and sculpted with a flat screwdriver.

Fasteners and fixing

Screws, clamps, nuts and bolts; known as fasteners, are a very important aspect of speaker building. Using the wrong ones and a slip with the screwdriver can result in a stabbed and torn cone. That will cost you £100 for a B110A. Use proper Allen head/hex socket screws as shown in Fig.3. Torx driver heads are even better. (If you use old-fashioned slotted head types don't say you haven't been warned!)

Generally, M4 bolts are used for Hi-Fi speakers. Some old ones made in the UK use 2BA bolts. I should just point out here that bolts or machine screws have a straight machined thread and fit into a corresponding threaded piece of metal. Wood screws and other self-tapping screws have a pointed end with a tapered thread of much coarser pitch.

Cheap speakers just use wood screws screwed into the baffle material to mount the drivers, but they usually come loose, causing rattles, especially if they are repeatedly screwed in and out. The professional approach is to put a threaded metal insert into the wood, such as a T-nut shown in Fig.4. This has to be inserted into the rear of the panel and the spikes hold it into the wood. This can be done with a hammer from the rear or pulled in with a screw and washer from the front (Fig.5).

Non-magnetic stainless steel (type A2) or brass screws are preferred because they don't shed magnetic particles (small shards of steel) while being screwed in. This can contaminate the magnetic gaps of the drivers. A scratching iron-filing filled voice coil is the worst possible form of audio distortion. I remember scolding an operative in the cabinet shop for using wire wool. While this is a normal method for surface finishing in carpentry, with speakers, the little steel fibres can enter the driver through the rear vent and spider.

It is always a good idea to use plain washers under bolt heads and nuts to protect the driver's frame from scratching. For the LS3/5A, shake proof washers were originally used to lock the nuts. I prefer to use the modern self-locking nuts called 'Nylocs', as shown in Fig.6.

Occasionally, special speaker clamps are used to mount drive units by clamping the rim, as shown in Fig.7. They

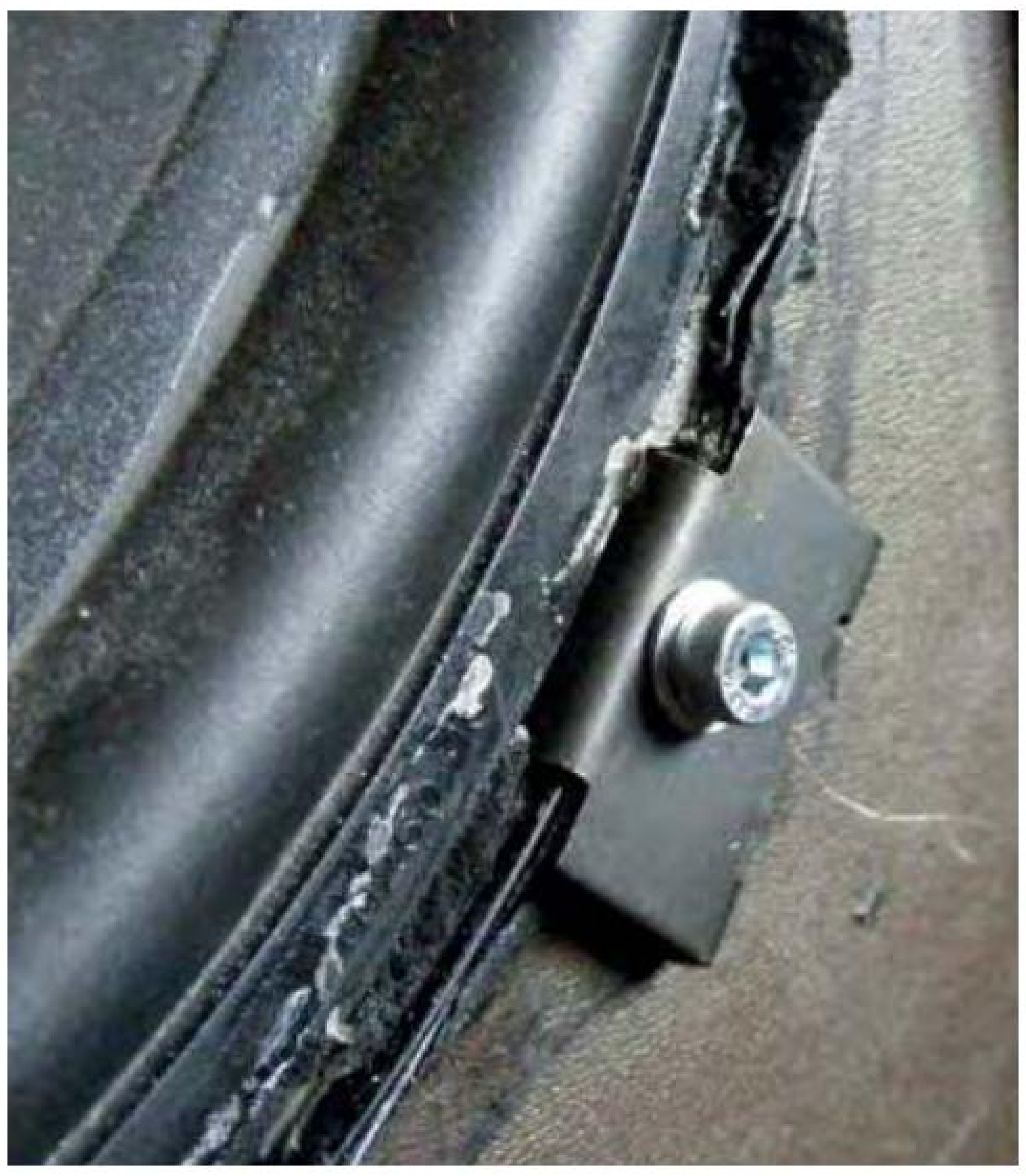


Fig.7. Specialised speaker clamps are available for rim clamping drivers.

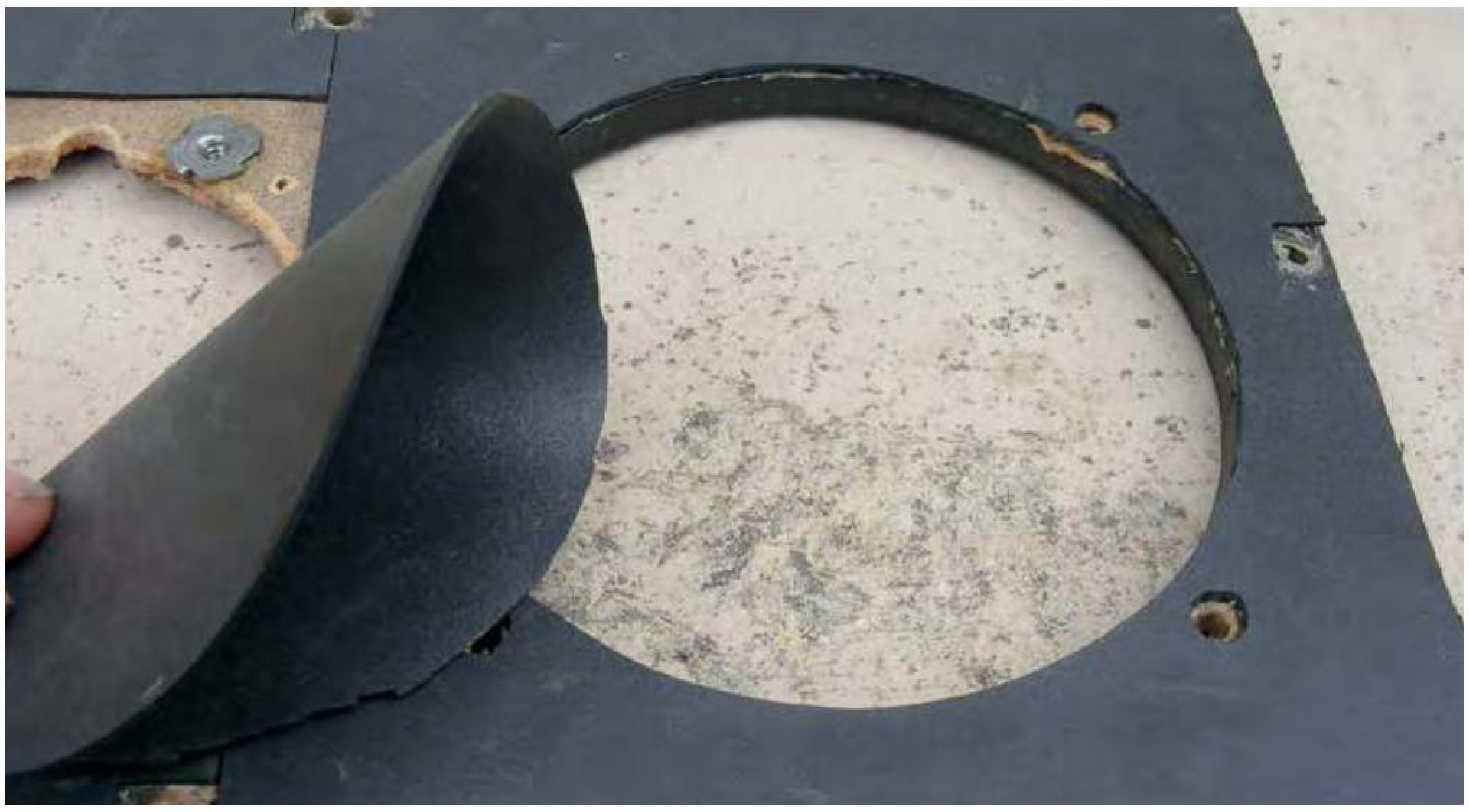


Fig.8. Sticking Neoprene rubber sheet to an LS3/5A baffle to give good sealing and chassis damping for the B110 woofer.

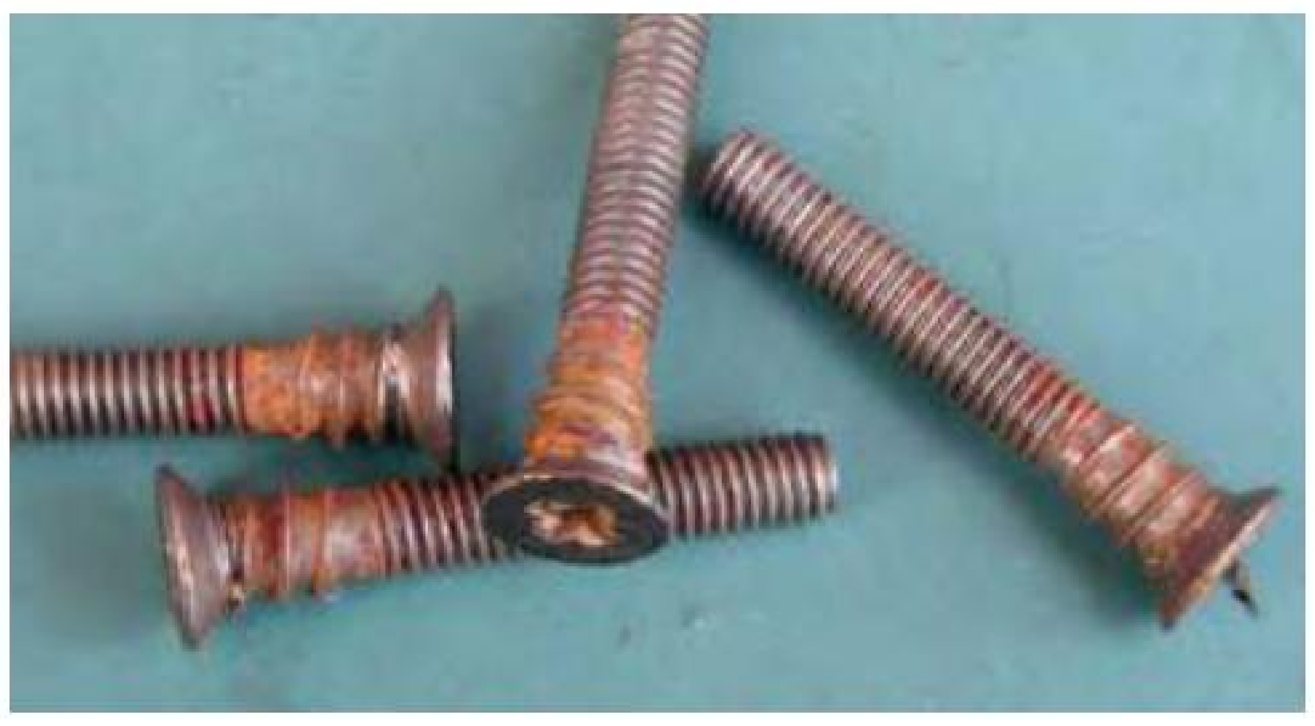


Fig.9. Anchor screws enable a bolt to be fixed to wood. They don't come undone when the nut is tightened since the wood thread is reversed.

have the advantage of allowing the mounting holes to be further away from the edge of the driver opening. This allows large T-nuts and torques to be used without edge breakage.

LS3/5A driver mounting

Unfortunately, the LS3/5A bass driver is rear mounted, which means air leaks are much more likely to occur. Special PVC edging strip is available to put on the front edge of the B110A frame, which also damps any metallic ringing. My approach is to glue 1mm neoprene rubber sheet from a builders' merchant to most of the rear of the baffle. It goes right to the edge of the baffle board (as shown in Fig.8) to provide sealing when the baffle assembly is screwed to the box battens.

It is tricky to avoid air leakage through the B110A chassis bolt holes. The original LS3/5A specification calls for a specialised loudspeaker bolt called an anchoring screw. This has a reverse threaded wood holding section in addition to the normal machine screw 4BA or M4 thread. Some recycled ones are shown in Fig.9. These screws are difficult to get hold of now, so I fix a normal M4 bolt in with a nut as illustrated in Fig.10. There is then the problem of making an airtight seal with the fixing nuts for the driver frame. The original LS3/5A used two red coloured fibre washers under the nut, but again these seem hard to get now. I use small rubber grommets cut in half, which provide the sealing function and a degree of resilient mounting. This is shown in Figs 11 and 12. The T27 tweeter is mounted from the front with T-nuts as shown in Fig.13. Very thin foam is normally used as a gasket for the tweeter. Usually this is supplied with the driver (see Fig.14).

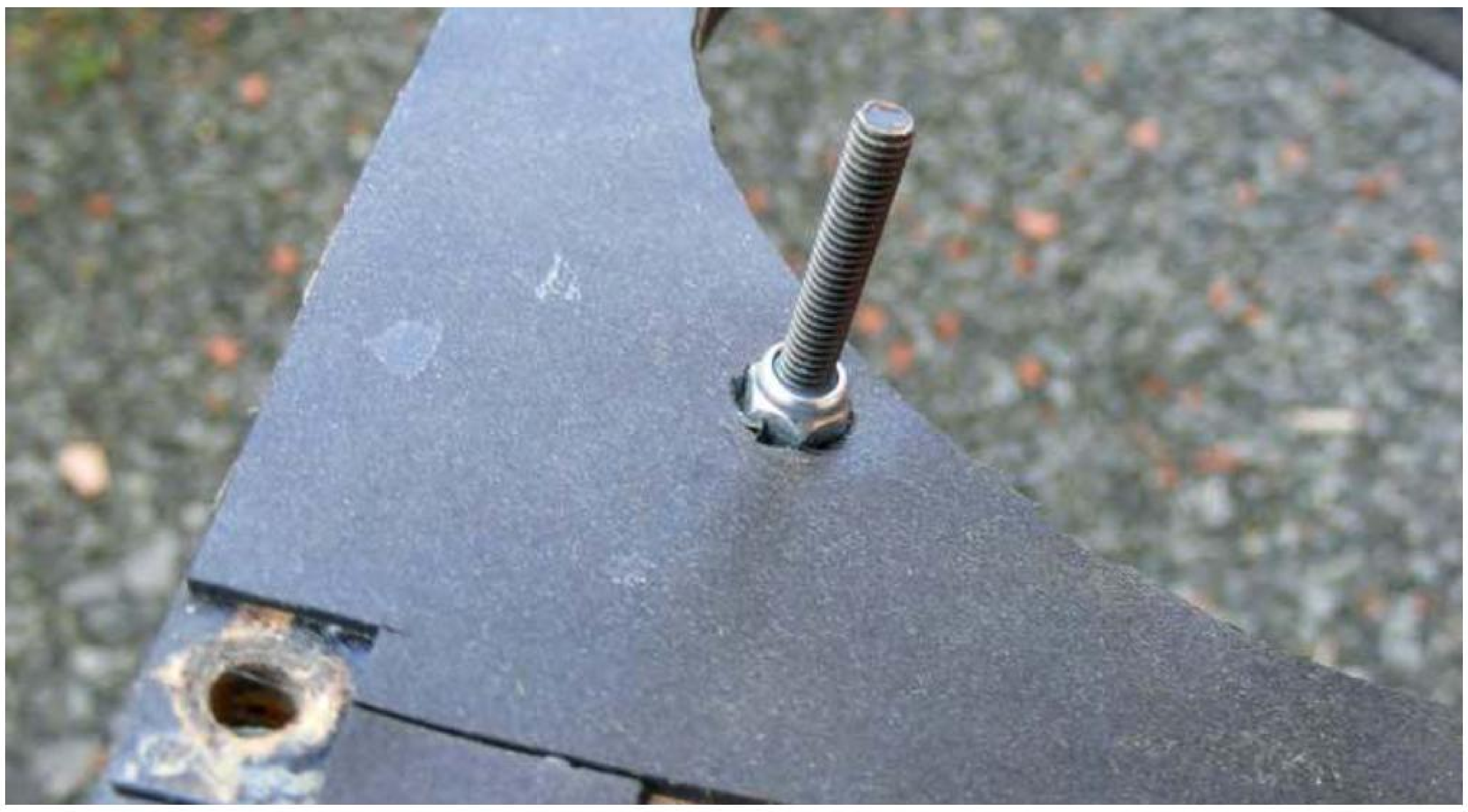


Fig.10. Although the original LS3/5A used anchor screws, I avoid them by just using a normal bolt and nut.

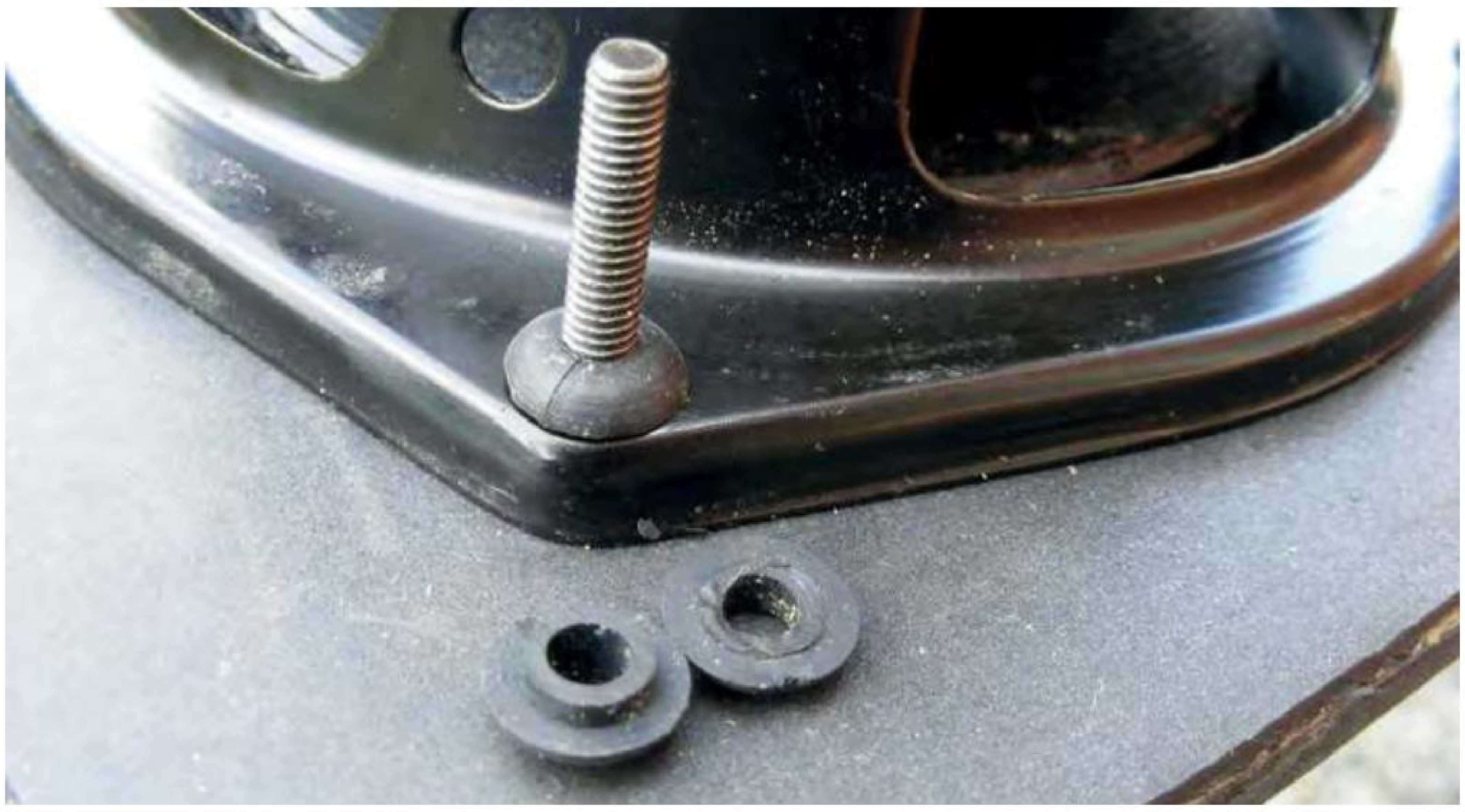


Fig.11. Sealing the mounting holes on the rear mount woofer in the LS3/5A can be difficult. Rubber washers or cable grommets cut in half provide a good solution.

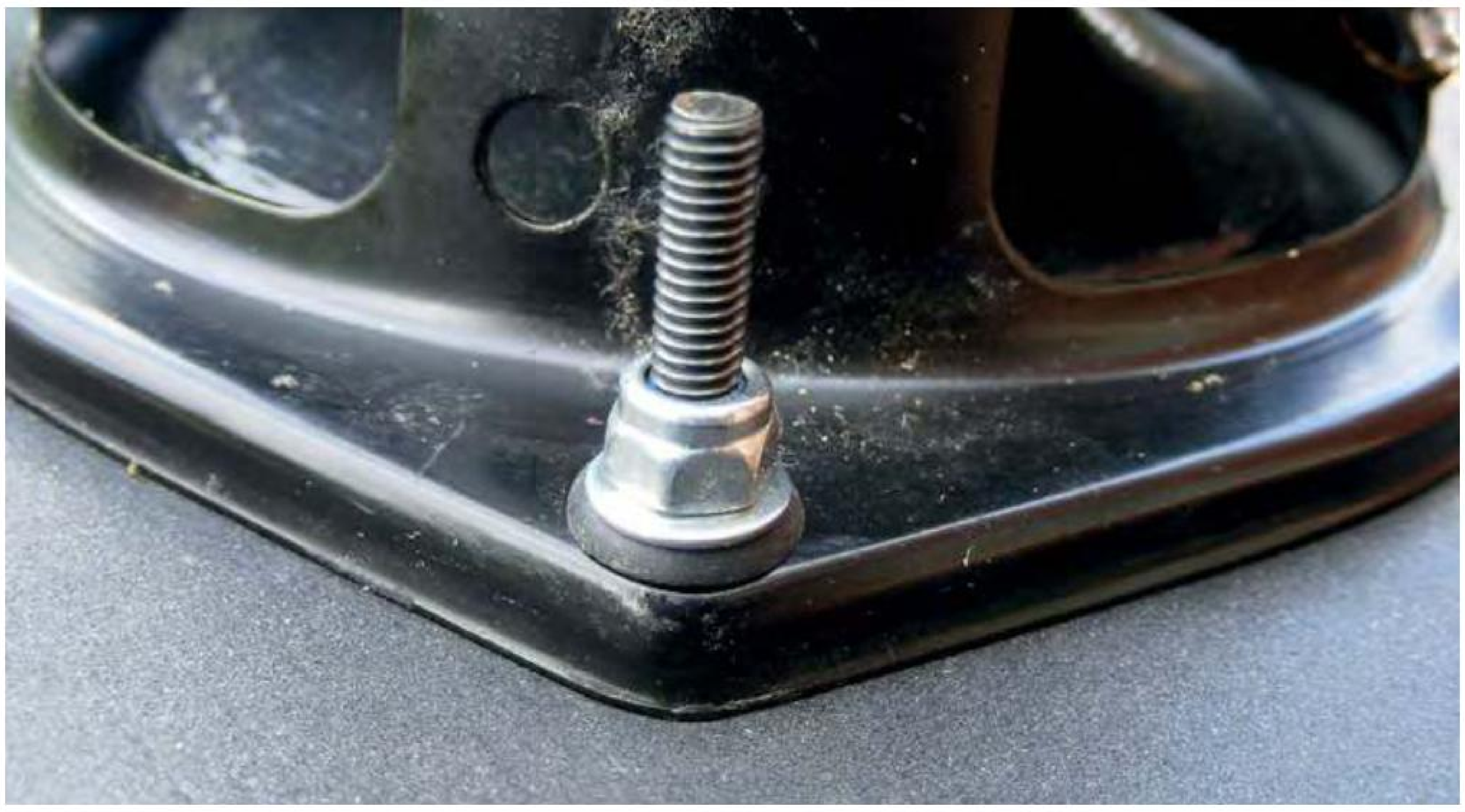


Fig.12. Final mounting arrangement for a B110. Note the Nyloc nut and plain washer to prevent chewing of the grommet during tightening.

Screw the box

Wood screws (with glue) are used to assemble the box. I use countersunk-head Pozidrive types, which means the heads are flush with the wood and can be painted over. Use a plated finish to avoid rusting. Drill a 3mm countersunk clearance hole in the panel and a 1.5mm pilot hole in the beech battens. This avoids splitting when everything is tightened up.

Damping

A great deal of fuss is made about electrical damping in loudspeakers, and the term 'damping factor', the ratio between speaker impedance and source impedance, is bandied about as if it was of great importance. So, if we have an 8Ω speaker fed by a source impedance of 0.1Ω via the amplifier and cable, the damping factor is said to be 8/0.1 = 80. In reality, this is nonsense because the real damping factor will be limited by the DC resistance of

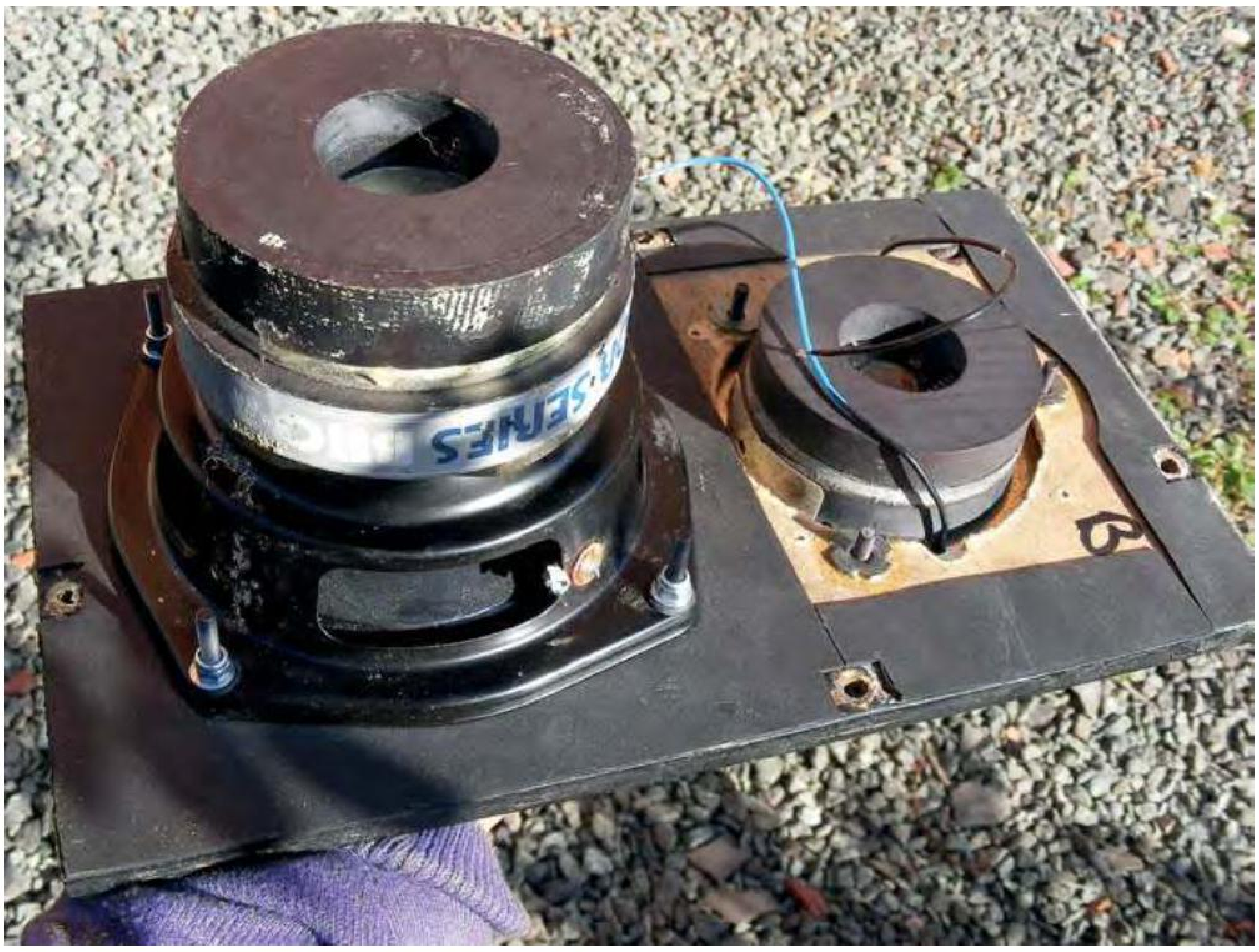


Fig.13. T-nuts have been inserted into the baffle for the T27 tweeter. A touch of glue has been applied to stop them being pushed out, which happens if too much downwards pressure is applied when tightening the bolts. (Note the extra screening magnets glued to the drivers – to be discussed in another article!).

the voice coil (typically 6Ω to about 1.33Ω), which is sufficient to absorb the back electromotive force (EMF) from the speaker to damp it adequately.

Panel damping

I've previously discussed (*Audio Out*, *PE*, September 2019, p.51-2) acoustic damping and the 'knock test' for cabinet walls, which should emit no sound (like a brick) when rapped with the knuckles. It's important to understand that this is *not* the kind of acoustic damping that can be achieved with light fluffy material. Cabinet damping requires mass – it must be heavy, which is why the following techniques are used.

My preferred material for cabinets is bituminous damping pads, but they are difficult to source. It is possible to use readily available car body damping sheets made by companies like 3M and sellers of generic materials on Amazon, for example: **https://amzn.to/2maz5X6**. This box of four 500mm × 500mm sheets costs £23.00, enough for one pair of small speakers, such as LS3/5As.

These panels are generally only 2mm thick, so it is necessary to stick three or four layers together, holding them down with panel pins or staples. The self-adhesive qualities are good enough for smooth metal, but for rough wood surfaces extra glue is needed – eg Thixofix.

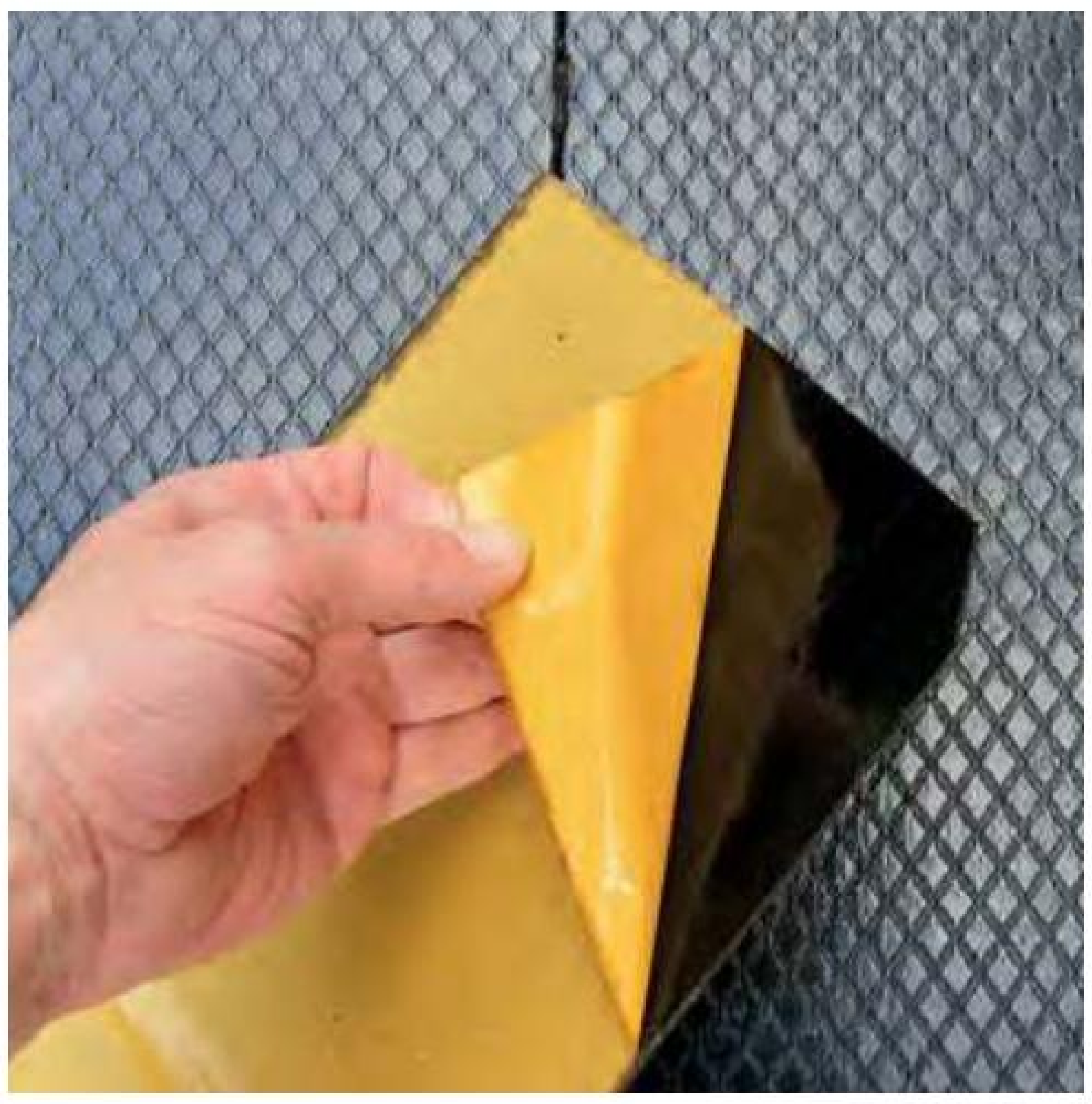


Fig.15. Car-body damping sheets – they have to be stuck together to make a proper speaker damping pad.

The dimples on the surface can also cause lack of adhesion. This surface finish can be seen in Fig.15. Another product I have found that works well is electrical insulation putty, such as 3M Scotchfil. The standard thickness is 6mm, so two layers is a good idea. All bituminous products adhere better when heated with a hair drier or hot-air gun.

Messy business

I make my own damping panels very cheaply by binding together five layers of soft unsanded roofing felt with Aquaseal 88 bituminous roofing sealant (shown in Fig.16). I spread the 'gunge' with a piece of metal plate, but it is a very messy process that has to be done in a shed or outside (Fig.17). Sunny warm weather helps keep the felt sheets soft and flat, and aids the drying process. After a week, the damping panels can be cut to size with a Stanley knife and metal straight edge (Fig.18). It's a good idea to sprinkle a bit of paraffin or Work Zone

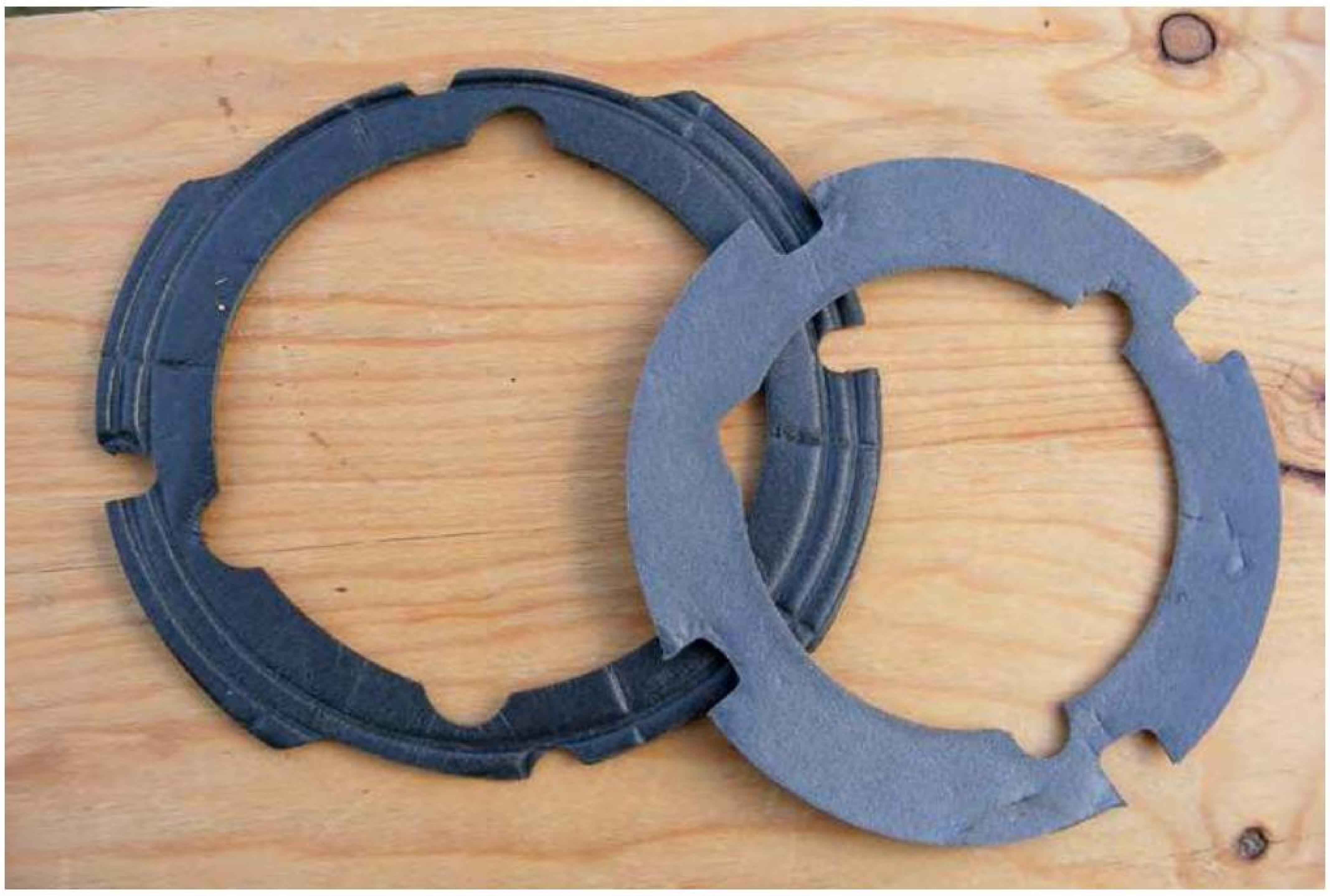


Fig 14. Foam gaskets are often supplied for drivers to provide effective sealing.

Fig.16. Aquaseal 88 is an excellent bituminous mastic for sticking roofing felt together to make damping panels. Keep away from white carpets, white cats… always use outside!

PoLabs
For more information or
software download please visit
www.poscope.com/epe

PoKeys
Connect, Control
- USB
- Ethernet
- Web server
- Modbus
- CNC (Mach3/4)
- IO
- PWM
- Encoders
- LCD
- Analog inputs
- Compact PLC

Stepper motor drivers
Drive
- up to 256 microsteps
- 50 V / 6 A
- USB configuration
- Isolated
- up to 32 microsteps
- 30 V / 2.5 A

PoScope Mega1+
PoScope Mega50
Measure
- up to 50MS/s
- resolution up to 12bit
- Lowest power consumption
- Smallest and lightest
- 7 in 1: Oscilloscope, FFT, X/Y, Recorder, Logic Analyzer, Protocol decoder, Signal generator

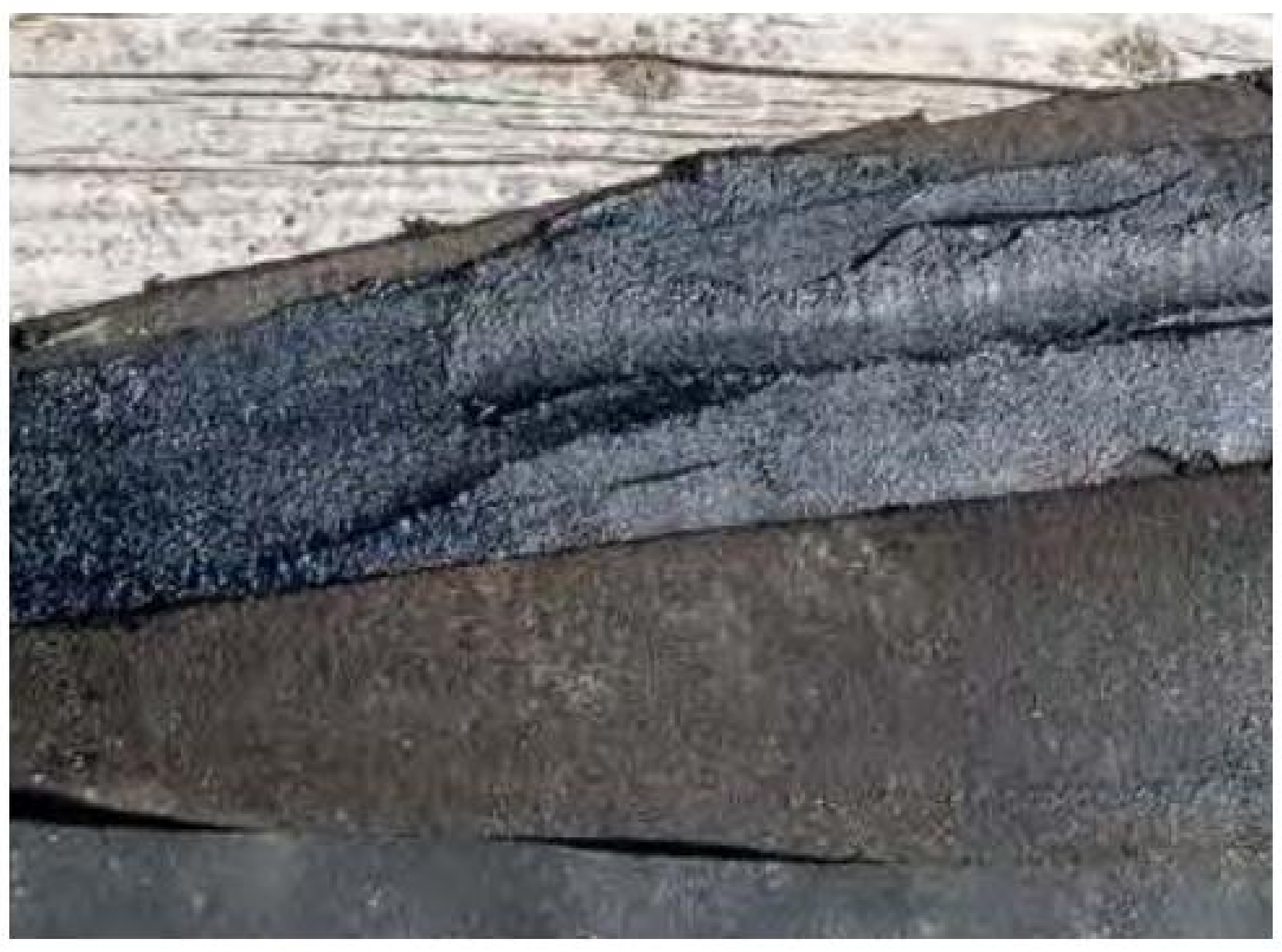


Fig.17. Coating the felt with mastic to make a multilayer damping sheet.

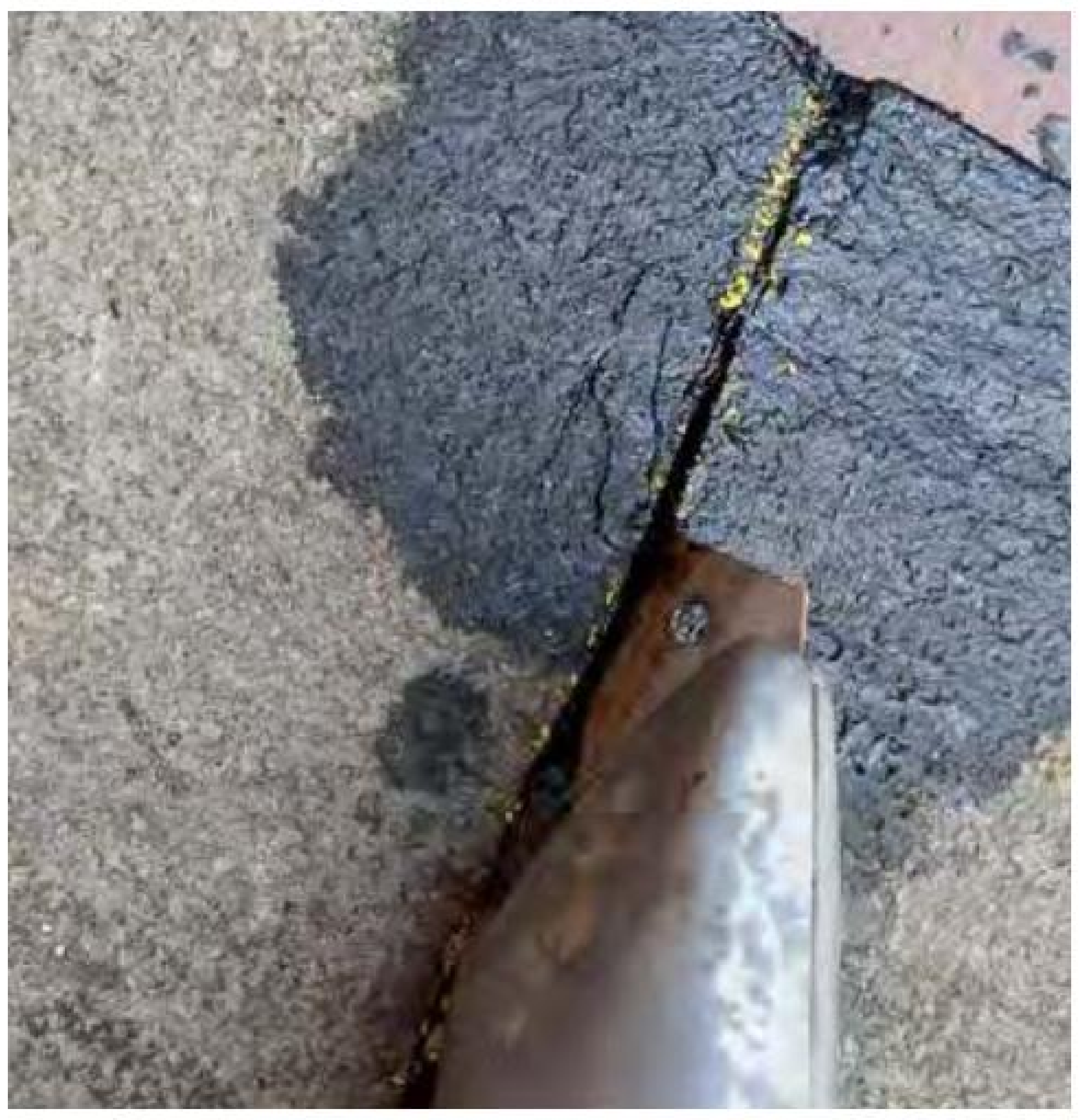


Fig.19. Lubricating the knife blade stops it getting stuck; it's a sticky cake.

contractors solvent from Aldi's along the cutting line to lubricate the knife and stop it getting stuck (Fig.19). When the panels are finished, attach them to the cabinet panels with a generous coating of Aquaseal and secure them with six 19mm (¾-inch) panel pins (Fig.20). Only use six pins – the shear action between the wood, Aquaseal and panel aids the damping and too many pins restrain it.

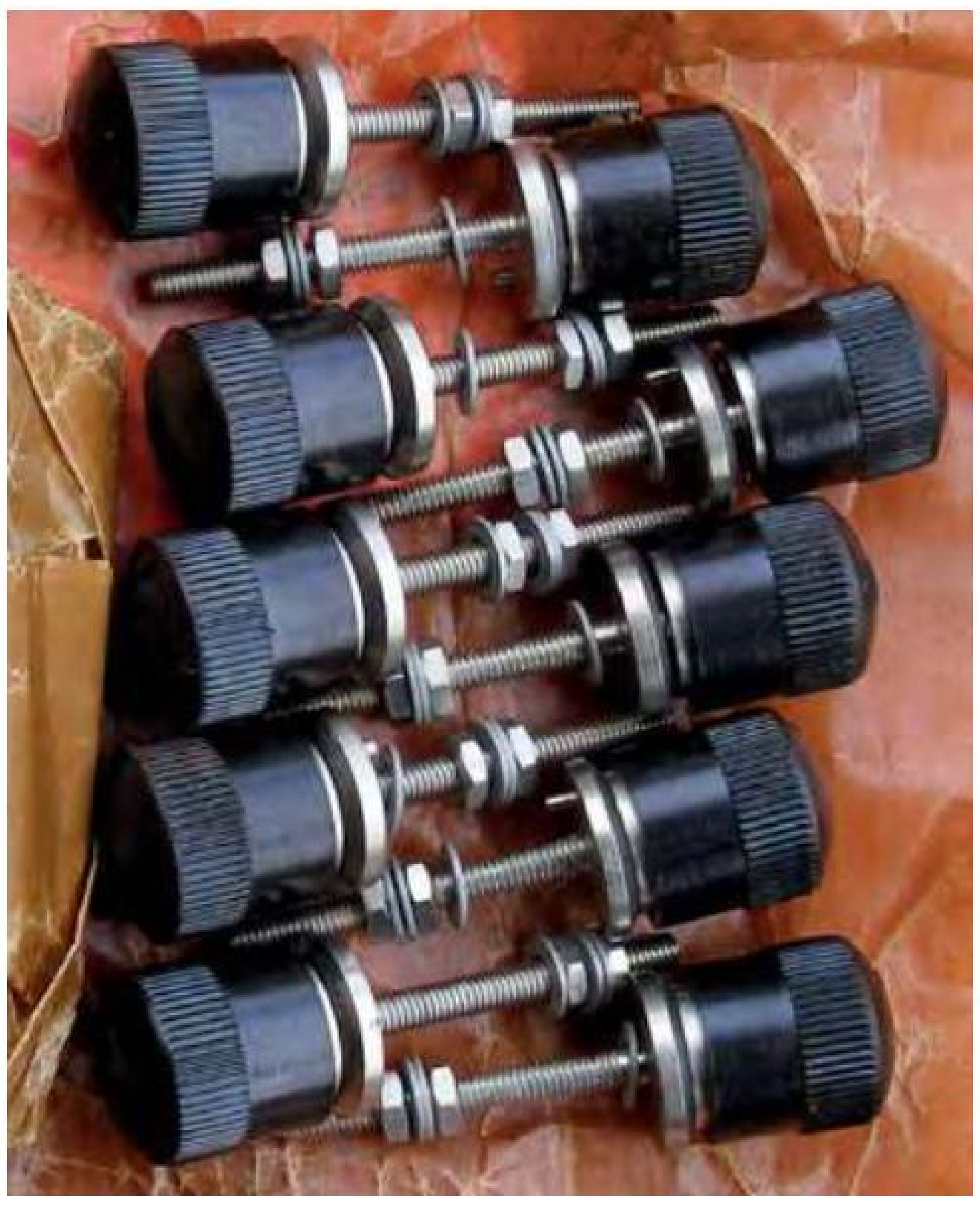


Fig.21. These extra long screw terminals from Birkett's (01522 520767) can go right through the wood and damping pad. We'll cover speaker connectors next month.

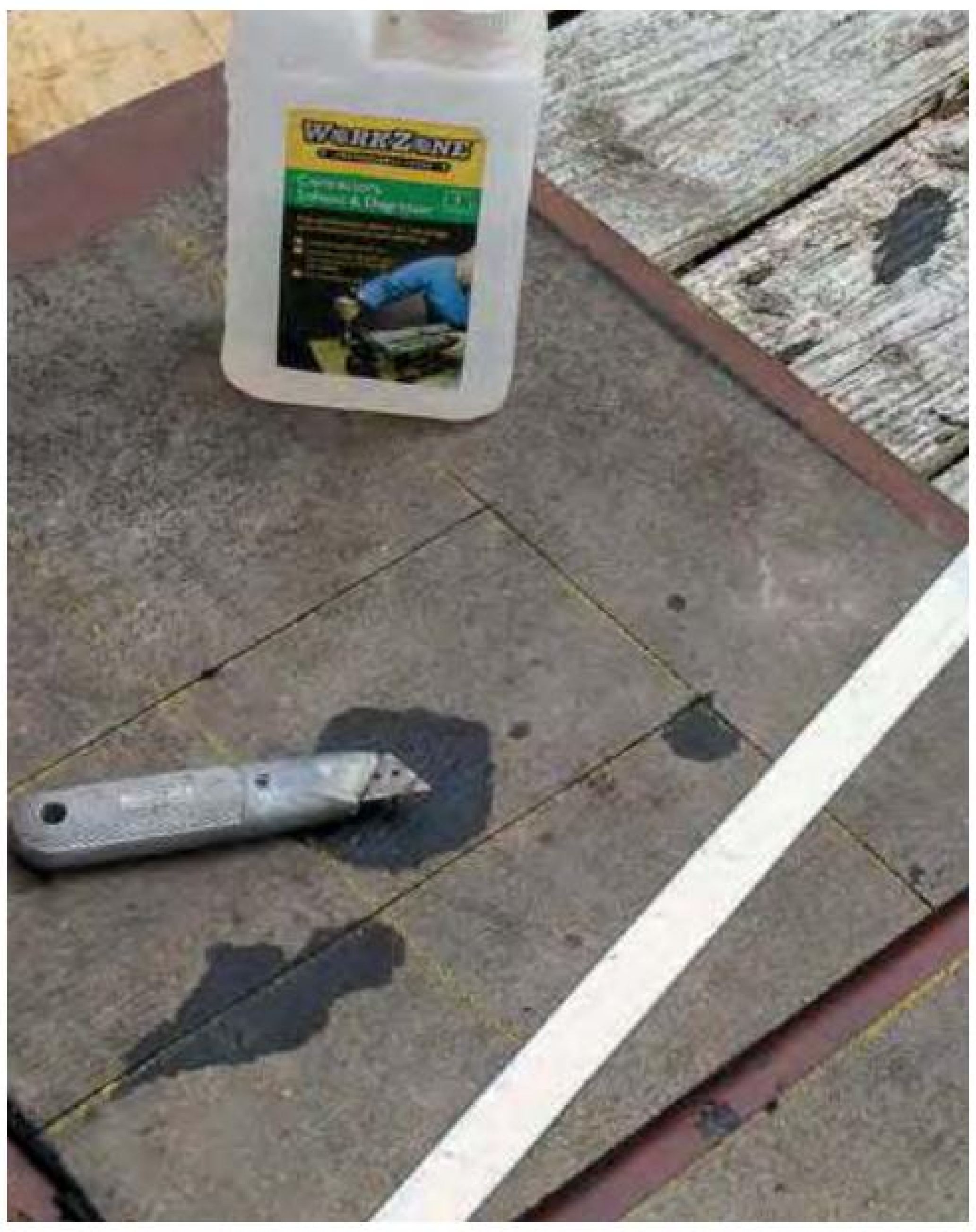


Fig.18. After the 'sandwich' has dried it can be cut using a Stanley knife and metal straight edge.

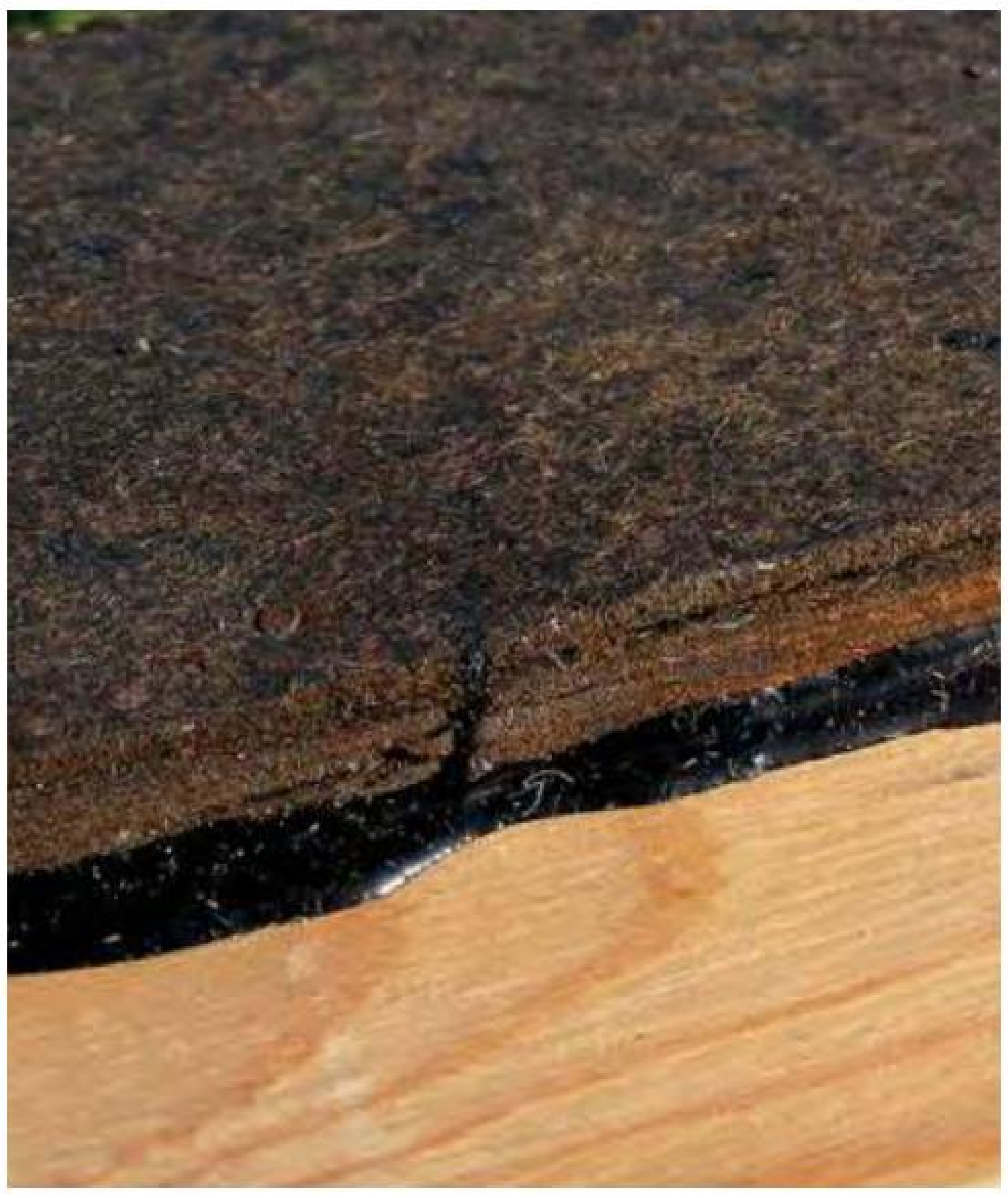


Fig.20. Damping pad attached to panel. Note liberal coating of Aquaseal to stick it.

Always attach the damping pads *before* assembling the box. It is just possible to do it after the box has been constructed, but it's difficult to hammer the pins in from inside and a messy job becomes even messier. In the factory we used a pneumatic industrial staple gun, but U-shaped staple nails can be used instead. Overall, this panel making technique is not much more involved than icing a cake, but only worthwhile if you make speakers and studio furniture frequently. It is what I would recommend because it is nearest to the specifications for the LS3/5A's original Bostik damping pad material.

On the original LS3/5A, the rear panel is undamped, but I always damp mine. The pad can get in the way of the connectors, but a lot of screw terminals are long enough to go right through the wood and damping panel as shown in Fig.21.

Next month

We'll conclude this diversion from LS3/5A's next month with a look at speaker connectors and wiring.

Practical Electronics

BACK ISSUES

EVERYDAY PRACTICAL ELECTRONICS

NOVEMBER 2018

PROJECTS • Super-7 AM Radio Receiver – Part 1 • 6GHz+ Touchscreen Frequency & Period Counter – Part 2 • High-Performance Dipole Loudspeaker • Banggood's £16 100kHz-1.7GHz Build-It-Yourself SDR Kit • Dynamo Bicycle LED Lighting System

FEATURES • Techno Talk • Net Work • Lucy's Lab • Circuit Surgery • Audio Out • Electronic Building Blocks

DECEMBER 2018

PROJECTS • Touchscreen Altimeter • Super-7 AM Radio Receiver – Part 2 • 6GHz+ Touchscreen Frequency & Period Counter – Part 3 • Using Cheap Asian Electronic Modules – Part 11

FEATURES • Techno Talk • Net Work • Lucy's Lab • Teach-In 2019 – Powering Electronics – Part 1 • PiC n' Mix • Circuit Surgery • Audio Out • Electronic Building Blocks

JANUARY 2019

PROJECTS • 1.5kW Induction Motor Speed Controller – Part 1 • Altronics Mega Box • Using Cheap Asian Electronic Modules – Part 12

FEATURES • Techno Talk • Net Work • Teach-In 2019 – Powering Electronics – Part 2 • PIC n' Mix • Circuit Surgery • Max's Cool Beans • Audio Out • Electronic Building Blocks

FEBRUARY 2019

PROJECTS • 1.5kW Induction Motor Speed Controller – Part 2 • Arduino Mega Box Music Player • MIDI Ultimate Synthesiser – Part 1 • Using Cheap Asian Electronic Modules – Part 13

FEATURES • Techno Talk • Net Work • Teach-In 2019 – Powering Electronics – Part 3 • Audio Out • Circuit Surgery • PIC n' Mix • Make it with Micromite • Max's Cool Beans • Electronic Building Blocks

MARCH 2019

PROJECTS • 10-LED Bargraph • Earthquake Early Warning Alarm • Using Cheap Asian Electronic Modules – Part 14 • MIDI Ultimate Synthesiser – Part 2

FEATURES • Techno Talk • Net Work • Teach-In 2019 – Powering Electronics – Part 4 • Audio Out • Circuit Surgery • PIC n' Mix • Make it with Micromite • Max's Cool Beans • Electronic Building Blocks

APRIL 2019

PROJECTS • Internet-based 'GPS' Time Source • Infrared-sensing Heater Controller • Using Cheap Asian Electronic Modules – Part 15 • MIDI Ultimate Synthesiser – Part 3

FEATURES • Techno Talk • Teach-In 2019 – Powering Electronics – Part 5 • Net Work • PIC n' Mix • Circuit Surgery • Audio Out • Make it with Micromite • Max's Cool Beans • Electronic Building Blocks

MAY 2019

PROJECTS • 2× 12V Battery Balancer • Deluxe Frequency Switch • Using Cheap Asian Electronic Modules – Part 16 • MIDI Ultimate Synthesiser – Part 4 • USB Port Protector

FEATURES • The Fox Report • Techno Talk • Teach-In 2019 – Powering Electronics – Part 6 • Net Work • PIC n' Mix • Circuit Surgery • Audio Out • Make it with Micromite • Max's Cool Beans • Electronic Building Blocks

JUNE 2019

PROJECTS • Wide-range Arduino-based LC Meter • USB Flexitimer • Using Cheap Asian Electronic Modules – Part 17 • MIDI Ultimate Synthesiser – Part 5

FEATURES • Techno Talk • Teach-In 2019 – Powering Electronics – Part 7 • Net Work • PIC n' Mix • Circuit Surgery • Audio Out • Make it with Micromite • Max's Cool Beans • Electronic Building Blocks

JULY 2019

PROJECTS • Full-wave 10A Universal Motor Speed Controller • Recurring Event Reminder • Temperature Switch Mk2 • Using Cheap Asian Electronic Modules – Part 18 • MIDI Ultimate Synthesiser – Part 6

FEATURES • The Fox Report • Techno Talk • Teach-In 2019 – Powering Electronics – Part 8 • Net Work • PIC n' Mix • Circuit Surgery • Audio Out • Make it with Micromite • Max's Cool Beans • Electronic Building Blocks

AUGUST 2019

PROJECTS • Brainwave Monitor • Super Digital Sound Effects Module – Part 1 • Control your PC with an infrared remote • Watchdog Alarm

FEATURES • The Fox Report • Techno Talk • Net Work • Electronics for a car dynamometer • Circuit Surgery • Audio Out • Make it with Micromite • Max's Cool Beans • Electronic Building Blocks

SEPTEMBER 2019

PROJECTS • Intelligent Touchscreen Lathe Speed Controller • Twin Dipole Guitar Speaker • Cheap Asian Electronic Modules – Part 19 • Super Digital Sound Effects Module – Part 2 • White Noise Generator

FEATURES • Techno Talk • Net Work • Circuit Surgery • Audio Out • Practically Speaking • Make it with Micromite • Max's Cool Beans • Electronic Building Blocks

OCTOBER 2019

PROJECTS • Programmable GPS-synced Frequency Reference – Part 1 • Digital Command Control Programmer for Decoders • Opto-isolated Mains Relay

FEATURES • The Fox Report • Techno Talk • Net Work • Using Stepper Motors • Circuit Surgery • Audio Out • Make it with Micromite • Max's Cool Beans • Electronic Building Blocks

BACK ISSUES – ONLY £5.80

£5.80 per issue for UK incl p&p ■ £6.75 Europe Air Mail ■ £7.50 ROW Air Mail

We can supply back issues of PE/EPE by post.

We stock magazines back to 2006, except for the following:

2006 Jan, Feb, Mar, Apr, May, Jul
2007 Jun, Jul, Aug
2008 Aug, Nov, Dec
2009 Jan, Mar, Apr
2010 May, Jun, Jul, Aug, Oct, Nov
2011 Jan
2014 Jan
2018 Jan, Nov, Dec
2019 Jan, Feb, May, June

Issues from Jan 1999 are available on CD-ROM / DVD-ROM

If we do not have a a paper version of a particular issue, then a PDF can be supplied for the same price – your email address must be included on your order.

Please make sure all components are still available before commencing any project from a back-dated issue.

ORDER FORM – BACK ISSUES

☐ Back issues required (month / year) ....

Name ....

Address..

.... Tel: ....

Email ....

☐ I enclose cheque/PO to the value of £ ....

☐ Please charge my Visa/Mastercard £ ....

Card No ....

Valid From .... Card Expiry Date ....

Card Security Code .... (Last three digits on or under the signature strip)

SEND TO: **Practical Electronics, Wimborne Publishing Ltd**
113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU.
Tel: 01202 880299 **Fax:** 01202 843233.
Email: stewart.kearn@wimborne.co.uk **On-line Shop**: www.epemag.com

Payments must be in £ sterling – cheque drawn on a UK bank. All items are normally posted within seven days of receipt of order. Send a copy of this form, or order by letter / online if you do not wish to cut your issue.

Make it with Micromite

Phil Boyce – hands on with the mighty PIC-powered, BASIC microcontroller

Part 10: Adding fonts, measuring temperature, saving files

Last month, in Part 9, we discussed MMBASIC's graphical commands. In particular, we showed that by adjusting the various parameters of the `TEXT` command, we could alter the way characters looked on the IPS display. We showed how to magnify characters with the scale parameter. We also saw that if the built-in font is scaled up or down then it may result in the required characters being too big or small to fit nicely on the display. Also, sometimes, we might prefer to use a different style of font for certain information that is being shown; for example, to use 7-segment-style numbers for the digits on a clock. This month, we will show you how to add different size and style fonts to the one that is built into the Micromite.

That's the software side of this month's article. Turning to hardware, we are going to demonstrate how easy it is to make a digital thermometer, and will incorporate the temperature reading into our *MKC Clock* display. Since the IPS display is quite small, we'll add the ability for the IR remote transmitter to control whether the time and date, or the time and temperature is displayed. We will also enable the IR remote to adjust the brightness of the IPS backlight.

Extracting and saving code from a Micromite

First though, several readers have asked how to extract a program from the Micromite and save the code on a computer. So, if you have a customised program in you *MKC* that you want to come back to later, we will show you how to extract a program from the Micromite's program memory and save it to your computer as a .txt file.

Questions? Please email Phil at:
contactus@micromite.org

If you have been following this series, you will no doubt be familiar with how to download a program (ie, a .txt file) from the *PE* website and get the code into your Micromite – in other words, how to Load a program into the Micromite; in short:

1. Download the relevant .txt file and Save it on your computer
2. Open the .txt file in any text editor program (such as Notepad)
3. Highlight all the text (ie, the program code) with the mouse, or `Ctrl-A` (or equivalent)
4. Copy all text to the computer's clipboard with `Ctrl-C`
5. In your terminal application (such as TeraTerm), and at the Micromite's command prompt:
    - **i.** Type `AUTOSAVE`
    - **ii.** Paste clipboard (right-click, then 'OK' if using TeraTerm)
    - **iii.** Press `Ctrl-Z` to commit the program into your Micromite's program memory (do not use `Ctrl-C` as this aborts the `AUTOSAVE` process).

The above procedure will overwrite any program currently in your Micromite (remember that the Micromite can only have a single program in its program memory at any one time). So by following the above steps, you will overwrite the potential 'masterpiece' that you have created directly on the Micromite having used the built-in `EDITor`.

How to save a program

Here are the steps to 'extract' a program from the Micromite's program memory and Save it as a text file on your computer. In your terminal application (such as TeraTerm), and at the Micromite's command prompt:

1. Type `LIST ALL` (do not worry if this results in the program listing scrolling beyond the bottom of the screen.)
2. Highlight all the lines of code in the program that have just been `LISTed`. Be careful to only highlight the code and nothing else that may be on the terminal screen above or below the `LISTing`
3. (Note: In TeraTerm, the process of highlighting the text will also Copy it to the computer's clipboard and hence you can jump over the next step!)
4. Copy highlighted text (program) to the computers's clipboard
5. Now, on your computer:
    - **i.** Launch your text editor program (such as Notepad)
    - **ii.** Open up a new (ie, blank) text file
    - **iii.** Paste the computer's clipboard (the lines of code) into the text file
    - **iv.** Save the text file on your computer (use appropriate filename).

We recommend that you have a dedicated folder on your computer containing all your Micromite programs. That way it is easy to Load and Save any program between your computer and your Micromite. If you are working on a 'big' program, then Save it on you computer and also include a (sequential) version-number at the end of the file name. That way, if you make a mistake somewhere then you have the previous version to revert to.

One thing worth mentioning here – it is possible to use your computer to edit the program code directly in the text file (using your computer's text editor) and then upload it to your Micromite (using the `AUTOSAVE` method). This means you will already have a backup on you computer and hence won't need to extract it from the Micromite. However, as soon as you make a change to the program (on the Micromite, by using the Micromite's built-in `EDITor`), then clearly the Micromite's program will now differ from the text

Micromite code

The code in this article is available for download from the *PE* website.

file on your computer. Put another way, the Micromite's program and the computer's text file do not automatically synchronise with each other.

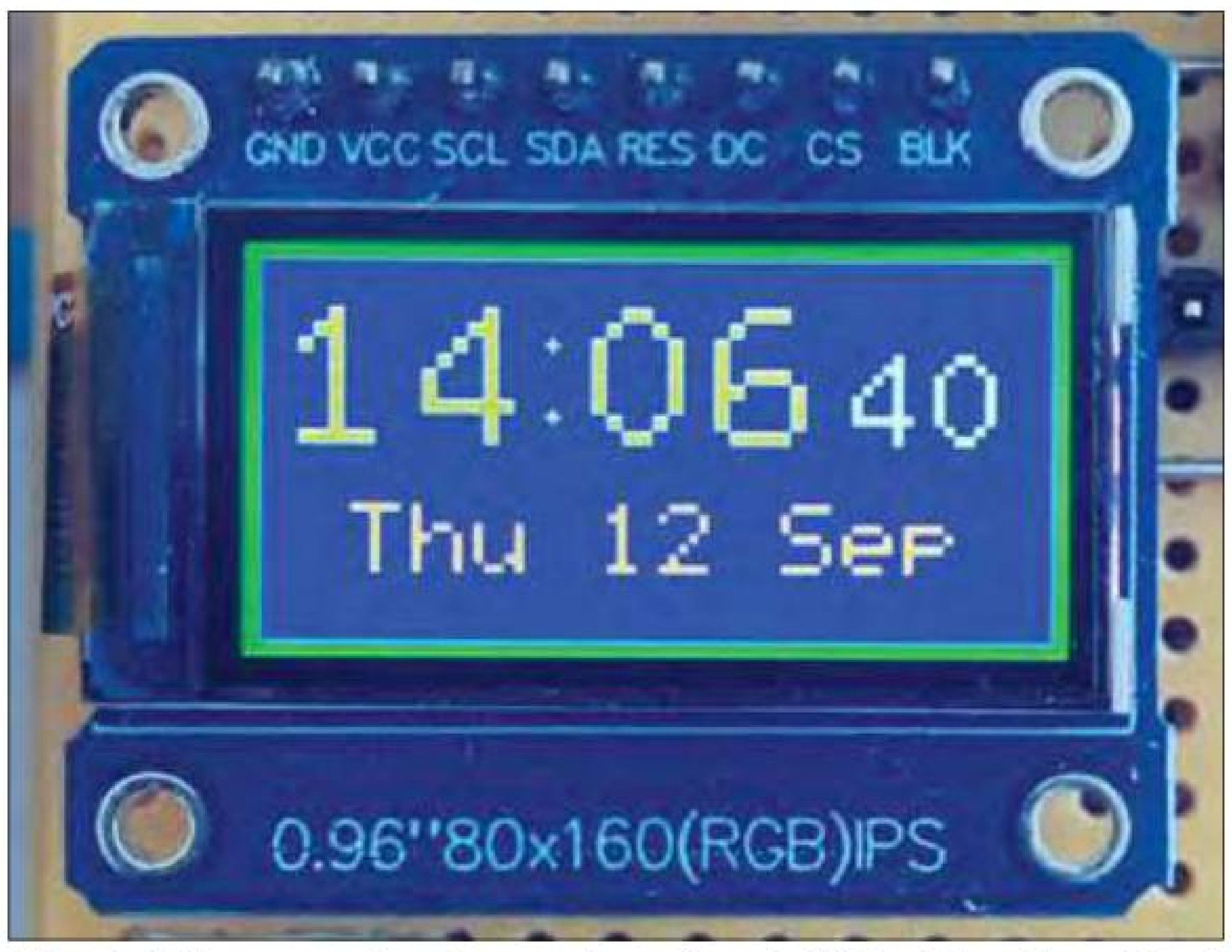


Fig.1. The result of running the MKC_Clock_v2.txt program. Note the date shows an abbreviated day-name and month-name rather than being totally numeric. The date is displayed here with a new font (Font 8).

The best thing to do is practise a little by creating a dummy program on the Micromite (using the `EDITor`), and then extract it to your computer using the procedure above (`LIST ALL`, Copy). Next, edit the text file by using a text editor on your computer. Then load the edited file back to your Micromite (`AUTOSAVE`, Paste), and launch the Micromite's built-in editor to ensure that you can see the changes that were made on the computer. After a couple of times, you will become very familiar with the whole process of Saving and Loading. As always, any questions – just drop me an email.

One final note regarding Loading and Saving programs. Neither the Load procedure, nor the Save procedure, will affect any code that you may have in the Micromite's Library (note that the Library should currently contain the IPS driver). Loading a program into the Micromite (with `AUTOSAVE`) will only overwrite the program-memory when `Ctrl-Z` is pressed and hence the Library code will remain in tact. Saving a program (with `LIST ALL`) will only list out the code contained in the program memory, and again, does not affect the Library in any way.

Adding fonts

Warning! This topic may appear to get rather involved at times – however, just take your time and work through it methodically at your own pace – it will be worth it. Understanding the concepts discussed here will allow you to create neat and elegant user-interfaces for your future projects. These concepts will also prove useful when it comes to using different sized screens.

If you followed last month's article then you may well have the **MKC_Clock_v2.txt** program still installed on your Micromite. If not, please download it from the October page of the *PE* website and load it into your Micromite; remembering first to extract and save any current code that you may want to keep (see above section). Run the v2 *MKC Clock* code and ensure that you see a display with a layout similar to that shown in Fig.1 (ie, with an abbreviated day and month description for the date – and which fits nicely on the screen).

Assuming you see the correct clock display, there are two things from the program that will now be discussed. Stop the program running (`Ctrl-C`), and launch the `EDITor` (F4). Now scroll down to the bottom of the code and you will see many hexadecimal numbers – similar to what the IPS screen driver looked like. Immediately above these hexadecimal numbers is a line of code: `DefineFont #8`, and below the numbers is a line of code: `End DefineFont`. This complete block of code simply defines the 'pattern data' for a custom font (which here has been declared as font 8). The hexadecimal numbers are generated by an external program (discussed shortly) – the main point here is to observe that there is a `DefineFont` / `END DefineFont` block of code which contains the relevant data to define a new font. With this new font, we are now able to refer to it with the font-number parameter (in any `TEXT` command) as font 8, as an alternative to the single built-in font (Font 1).

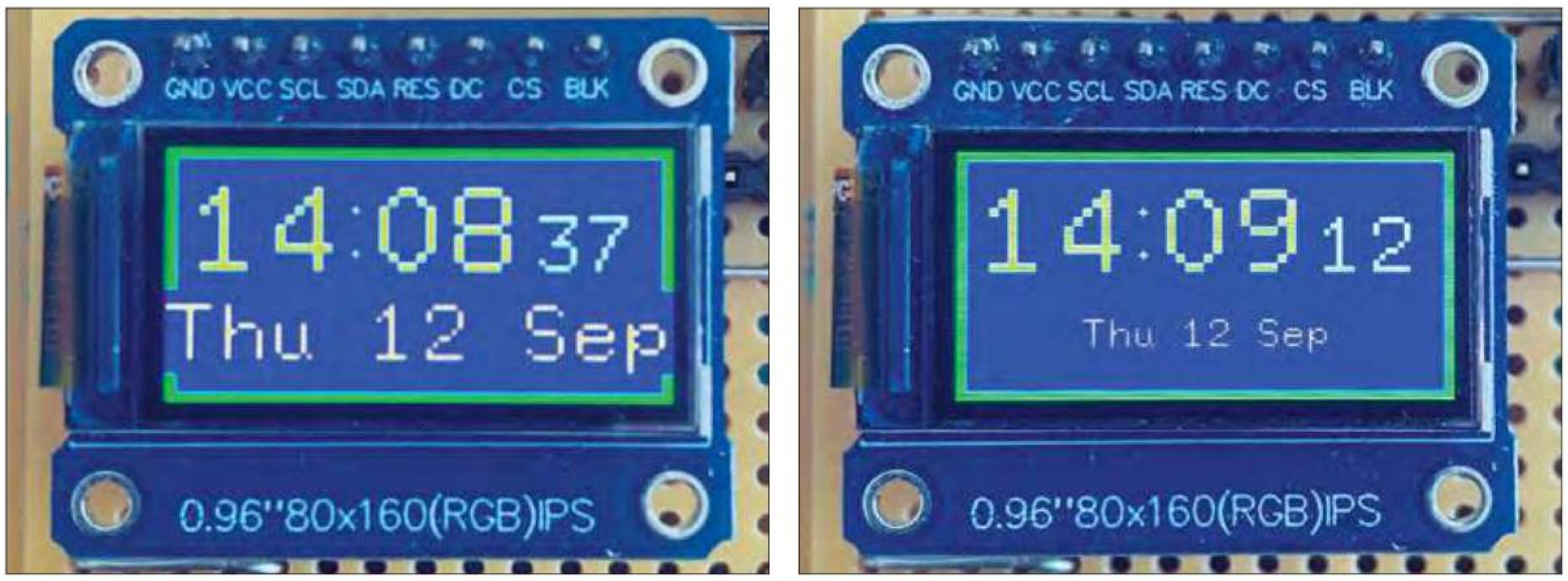


Fig.2. The result of using the built-in font (Font 1) for the date: a) (left) With a font-scale factor of 2, the date is just a bit too big to fit within the green border and b) (right) reducing the font-scale to 1 results in a date that fits within the border, however, it is a bit too small to read comfortably

Now scroll up to the last line of code in the main `Do`/`Loop` – the `TEXT` command is repeated here; it simply prints the date (contained in the string `DispDate$`) underneath the time:

```
Text 80,58,DispDate$,cm,8,2,
RGB(brown),RGB(blue)
```

Looking at the parameters, you will see it is indeed using the additional font (font 8), and with a scale of 2. Now let's edit the value 8 and change it to a value of 1 (so that the date gets displayed with the built-in font instead). `RUN` the program again and observe that the date is now too big to fit nicely on the screen (see Fig.2a).

What if we reduce the scale factor from a value of 2 to a value of 1 to make the date 'smaller' in size? Go ahead and try it – you will see that the date is now just a bit too small to read comfortably (see Fig.2b). Before proceeding, ensure the font and scale parameters are set back to the values of 8 and 2 respectively – this will result in the clock display once again looking like that shown in Fig.1.

The above scenario is a good example of why we may need to define a new font. For our *MKC Clock* program we required a different sized font since the built-in font was unsuitable (regardless of scale factor, as seen in Fig.2); it just didn't fit the date nicely onto the IPS display. As we have seen, v2 of the *MKC Clock* code includes font 8, which is an ideal size.

Font size

Each character in a specific font-number can be considered as a block of pixels with a fixed width and height. Within this block, individual pixels are either 'on' or 'off', with the overall pattern defining how the resultant character will look. To understand the block size of any font there are two built-in system values that we can read: `MM.FONTWIDTH` and `MM.FONTHEIGHT`. These will return the pixel quantity for either the width or the height for the default font. Also relevant here is the MMBASIC command `FONT`, which simply sets the default font that is used if we were to omit the parameter from the `TEXT` command. At the command prompt, type the following as a single line of code:

```
FONT 1 : PRINT MM.FONTWIDTH, MM.FONTHEIGHT
```

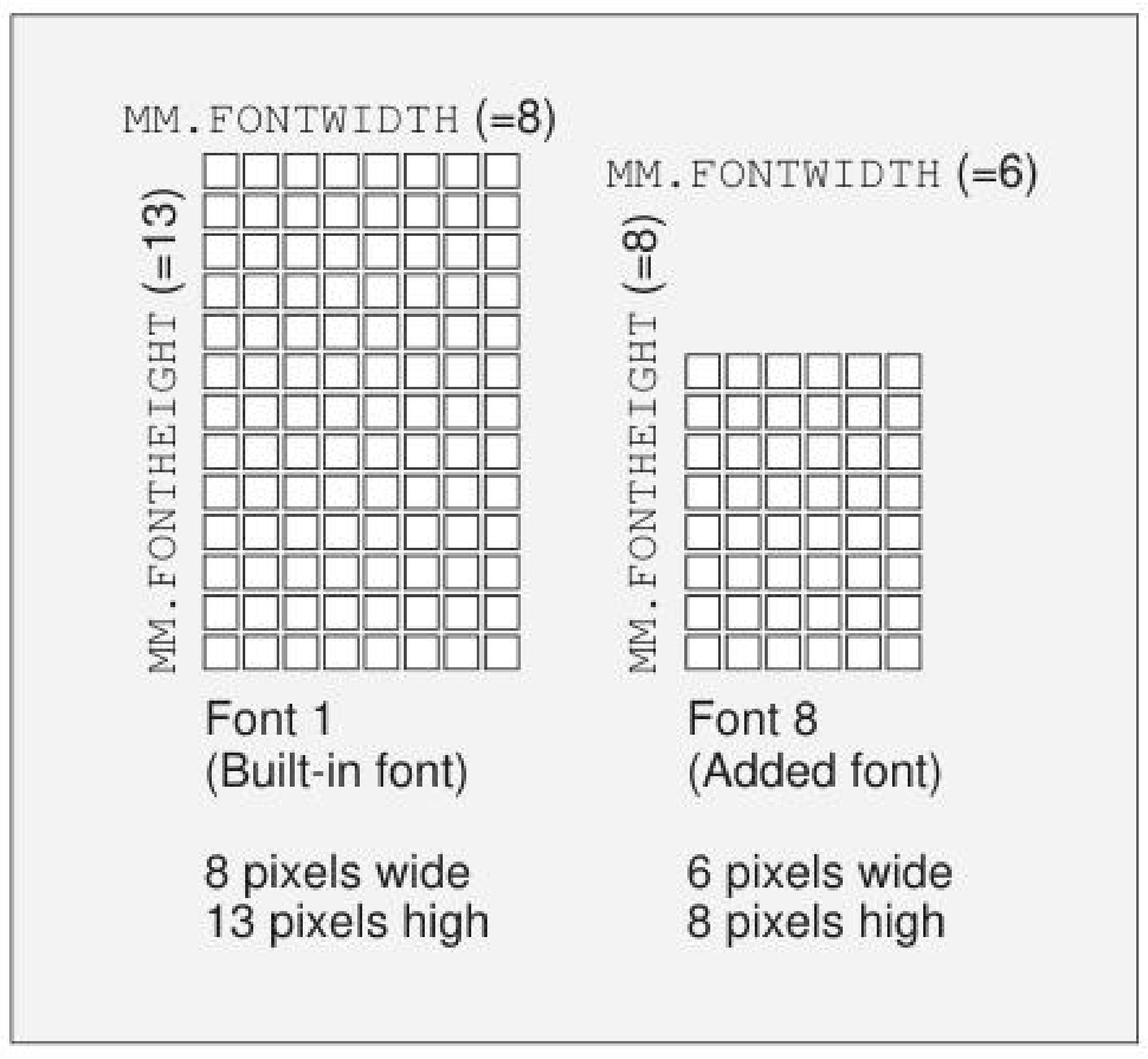


Fig.3. a) (left) The built-in font (Font 1) has 8x13 block-size and b) (right) v2 of the *MKC Clock* program adds a smaller font (Font 8) which has a 6 × 8 block-size

This selects `Font 1` as the default font, and then reports back the number of pixels defining the block size. Pressing `Enter`, you will see that `Font 1` (ie, the built-in font) has a width of 8 pixels, and a height of 13 pixels (refer to Fig.3a). Now let's consider the number of characters used to display the date in our clock program. Referring to Fig.1, you will see that there are three characters for the abbreviated day-name, then one for the space prior to the numeric date, two more for the numeric date, another one for the space, and finally three more for the abbreviated month – hence a total of 10 characters.

So, if we use `Font 1` for the date (remembering that `Font 1` has a width of eight pixels), then this means 10 × 8 pixels = 80 pixels. This calculation is based on a scale value of 1; however, we are using a scale value of 2, so this equates to two lots of 80 pixels. Hence, the 10 characters making up the date require a total of 160 pixels of screen width. The IPS display used in the *IDM* has a pixel resolution of 160 (wide) × 80 (high); so when displaying the 10-character date in `Font 1`, and with a scale factor of 2, (requiring a total of 160 pixels), it will only just fit on the display's horizontal resolution of 160 pixels (refer to Fig.2a to see how the date just fits edge to edge).

Now let's look at the block size for the font that has been added in v2 of the *MKC Clock* program. At the command prompt type the following:

```
FONT 8 : PRINT MM.FONTWIDTH, MM.FONTHEIGHT
```

You will see that the character block-size for `Font 8` is six pixels wide by eight pixels high (as depicted in Fig.3b). So our 10-character date, displayed with `Font 8`, and with scale factor 2, (as the v2 program uses) occupies a pixel-width of: 10 (characters) × 6 (pixels per character) × 2 (scale factor), which totals 120 pixels. This 120-pixel width fits nicely within the 160 horizontal screen pixels, and also within the green boarder.

You should now under-stand a little more about how the pixel resolution of the screen, the font block size, and the font scale factor are all interlinked and ultimately determine the size of the characters that get displayed on an attached screen.

Before we look at how to add font(s), we first need to understand two points. First, whether we're adding a pre-written font, or creating a new custom font, the end result will be a block of code (for each font added) that begins with the line, `DefineFont`, which is followed by lots of hexadecimal numbers, and finally ends with the line, `End DefineFont` (as we saw at the bottom of the v2 program code). From now on, in this discussion we will refer to this as the 'Font-Code'.

Second, it doesn't matter whether the Font-Code resides in program memory (ie, as part of the program itself), or is put securely into the Library (where it can't be accidently altered). Remember, putting Font-Code in the Library will compress it, freeing valuable space in the program memory, which means you can write bigger programs.

Assume for now that we already have the required Font-Code (we will come back to discuss this topic shortly). Let's summarise the steps to either insert the Font-Code into our program (as with the *MKC Clock* v2 program), or insert it into the Micromite's Library. The following may seem complex, but stick with it as there are several different scenarios that need to be covered.

Adding Font-Code

Scenario 1

If you are developing a program on the Micromite (with the built-in `EDITor`) and have code in the Micromite that you don't wish to lose, then follow these steps (effectively you are merging two .txt files into one on your computer, and then placing the result back into the Micromite):

1. Save (ie, extract) your Micromite program into a .txt file on your computer (ie, Save program using `LIST ALL`)
2. Open the required Font-Code in a text editor on your computer
3. Copy Font-Code onto your computer's clipboard
4. Now open your program code (from step 1) in a text editor
5. Paste the Font-Code after the very last line of your program code (everything now merged)
6. Save the resultant .txt file (optional – but recommended)
7. Select All from step 5, and Copy to the computer's clipboard (ie, program code which now includes the Font-Code)
8. AUTOSAVE, Paste, `Ctrl-Z` to commit to the Micromite's program to the memory.

Scenario 2

If you don't have any code in the Micromite, or you don't mind losing it, then simply perform steps 2, 3, and 8. This will result in just the Font-Code being in the Micromite; and from there you can add you code via the built-in `EDITor`.

Scenario 3

If you are developing a program in a text editor on your computer, then to add Font-Code just follow these steps (you are simply adding the Font-Code to the end of you program .txt file):

1. Open the required Font-Code in a text editor on your computer
2. Copy Font-Code onto your computer's clipboard
3. Now open your program code (if not already) in a text editor
4. Paste the Font-Code after the last line of your program code (everything now merged)
5. Save the resultant .txt file as the latest version of your program

Scenario 4.

Last, let's look at adding Font-Code into the Micromite's library. You would typically only do this once you are 100% happy with the font. If you have existing code in the Micromite that you don't wish to lose, then first extract it to a .txt file. (You can then reload it back into the Micromite after you have added the Font-Code to the Library). Now, to add the Font-Code to the Library:

1. Open the required Font-Code in a text editor on your computer
2. Copy Font-Code onto your computer's clipboard
3. `AUTOSAVE`, Paste, `Ctrl-Z` to commit the Font-Code to Micromite's program memory
4. `LIBRARY SAVE` (Font-Code now moved from program memory and compressed into Library). Now reload any program code using the usual `AUTOSAVE`, Paste, `Ctrl-Z`.

This should cover every potential scenario for adding the Font-Code into a project; but where do we get the Font-Code from in the first place? There are three main sources available to us:

1. Use Font-Code from a pre-written font file
2. Use Font-Code from a Micromite program listing created by someone else
3. Generate our own Font-Code by using an application that allows us to design custom fonts.

Do remember that each Font-Code in a program must have a unique font number for the line beginning: `DefineFont #x`, and it cannot be higher than 16 (and ideally not 1 – if you do use a value of 1, then the built-in font will be over written).

1. Font-Code from a pre-written font-file

This is easy; all you need to do is open up the font file in a text editor, and then simply copy the Font-Code onto the computer's clipboard. Then, use whichever one of the above four scenarios meets your situation to add the Font-Code from the clipboard.

Here's an example. To get some font files to experiment with, go to: **http://geoffg.net/micromite.html** and scroll to the Downloads section towards the bottom of the page. Click on the link to Download the Micromite Firmware, select Save, and from there, select the Embedded Fonts folder. In the Fonts folder you will see around 15 font files (as BAS files). Open any one of these in a text editor and you will see the Font-Code. At the start there will be a few lines of comments that show the font size (ie, block size) in pixels.

2. Font-Code from a Micromite program

This is also easy. Open the Micromite program (in a text editor) and simply Copy the Font-Code onto the clipboard. Then use as required. For example, open **MKC_Clock_v2.txt** in a text editor, and scroll to the Font-Code at the bottom of the listing. From there, Copy it to the clipboard.

3. Create a custom font

OK – this one involves a bit of work – but once mastered, it is lots of fun. To begin with, you will need to use a Windows machine so that you can run the required *FontTweak* program. This program allows you to create your own fonts and ultimately generate a (BAS) font file from which you can copy the Font-Code as outlined above.

Using the downloaded folder from **http://geoffg.net/micromite.html**, select the Embedded Fonts folder and then click on the *FontTweak* folder. You can then launch the application; however, it is recommended that you first read (and print) the three-page pdf file. This explains how to use the program. Limited space means we are unable to go into detail here, but in summary, *FontTweak* will ask for a starting ASCII character value (defaults to 32), and then ask you for the character block size. Once this information has been entered, you will be presented with a blank pixel grid for the first ASCII value you entered. You then use the mouse to turn on (left click) or turn off (right click) pixels in the grid. This allows you to form the pixel pattern for that particular ASCII character. You can then move onto the next character and define that pattern. Once you have defined all the patterns you require, simply select 'save as BAS' and it will create the required Font-Code for you. Please do take the time to explore this – it will allow you to create custom fonts which result in very smart user interfaces.

Rather than give step-by-step instructions on how to add another font to the existing v2 *MKC Clock* code, we have made a new version available for download from the November 2019 page on the *PE* website. Download the file **MKC_Clock_v3.txt** and load it into your Micromite. On running it, you should see pretty much the same as before, but there are now two additional fonts: font 8 is still the 'smaller date font', as used in v2, and the new font 7 is a 7-segment-style font used for the time digits. By the way, font 7 was created in the *FontTweak* program, and the resultant Font-Code has simply been copied into the v2 *MKC Clock* .txt file program code.

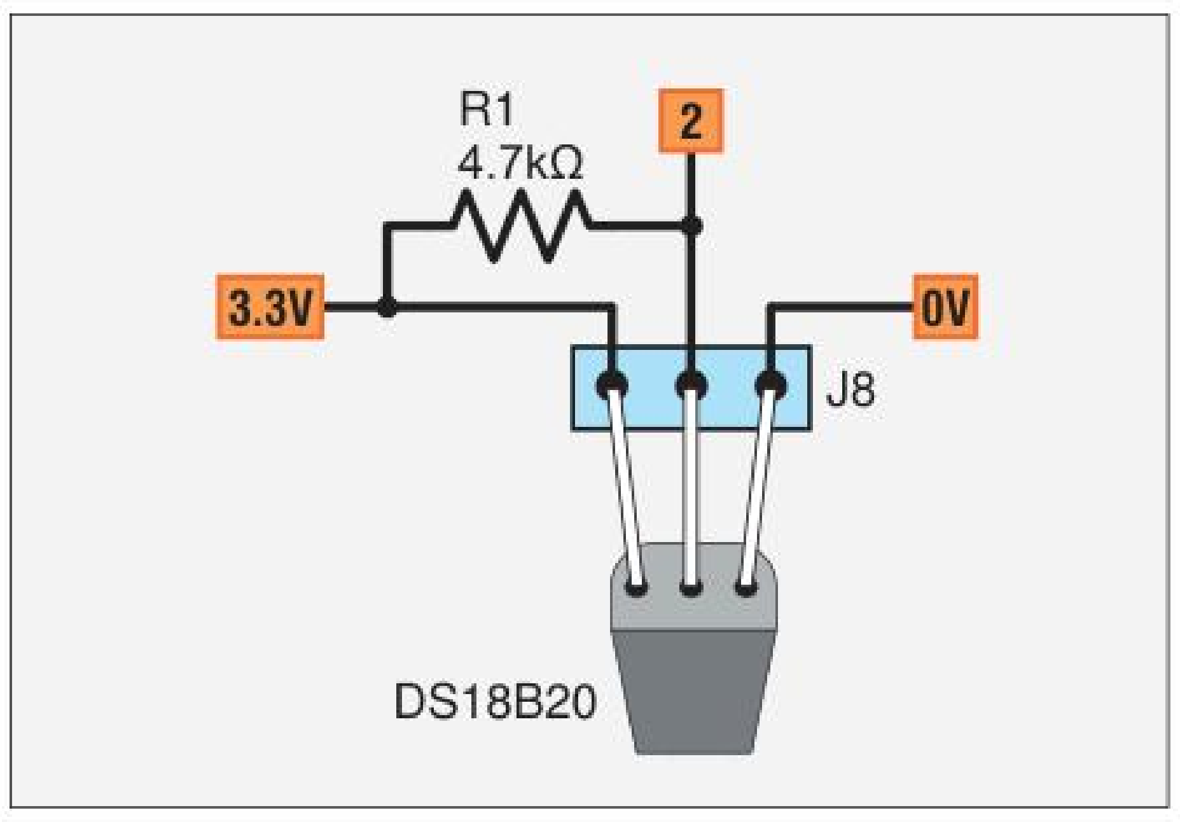


Fig.4. Circuit diagram for a simple Digital Thermometer.

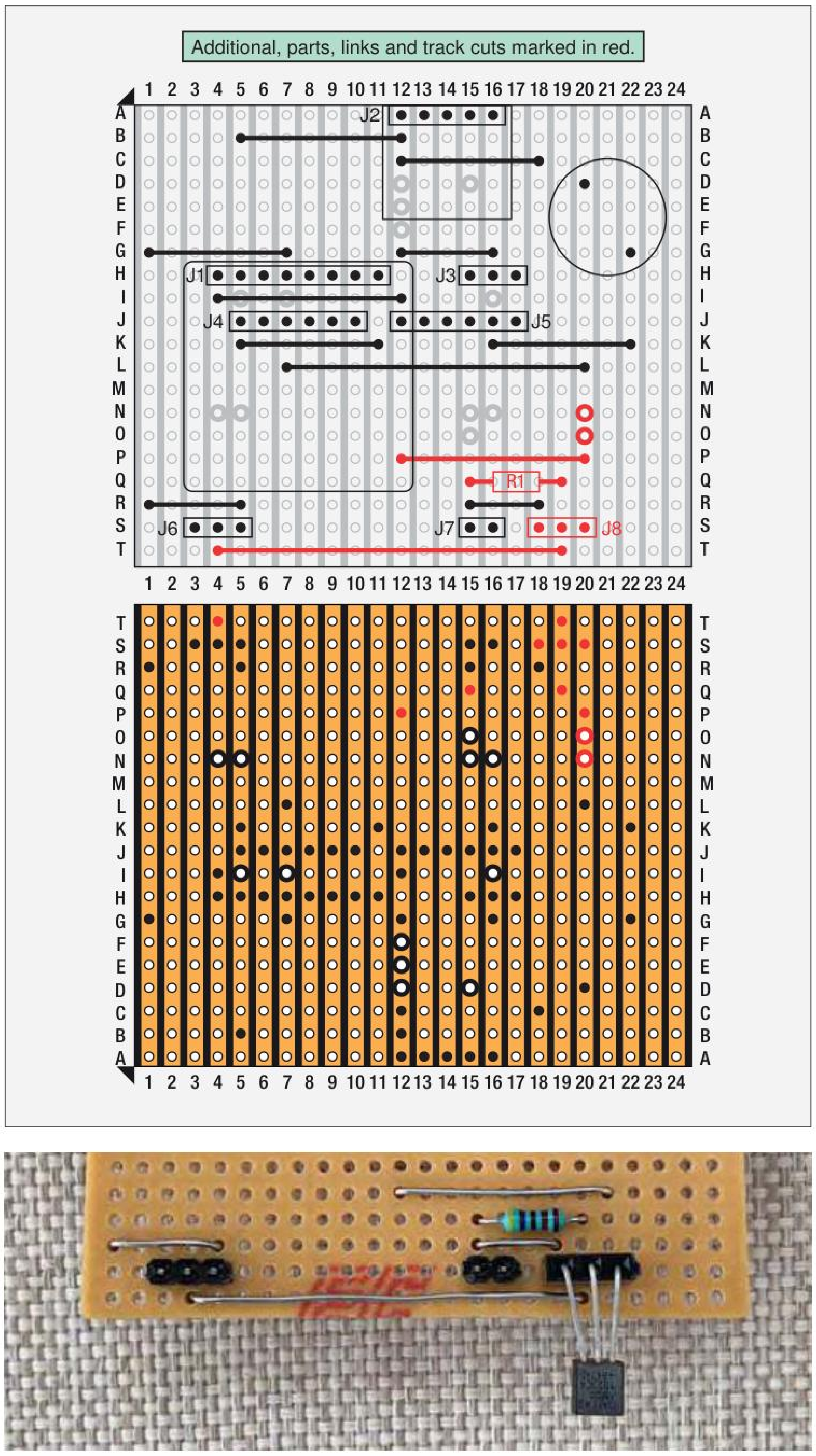


Fig.5. Modifications required to the IDM to incorporate a DS18B20 temperature sensor.

I hope this has provided a useful guide to using fonts. It is quite involved, so work through it at your own pace.

Measuring temperature

With the ability to display information on the IPS display in whatever format we need, it is now extremely easy to create a digital thermometer. The simplest way to measure temperature on the Micromite is to use the popular DS18B20 sensor, which allows the use of MMBASIC's built-in `TEMPR` command (the only other hardware required is a 4.7kΩ resistor). Take a moment to refer to the *Micromite User Manual* and read *Measuring Temperature* in the *Special Device Support* section – Page 26 in version 5.05 of the *Manual*.

The syntax for the `TEMPR` command is: `TEMPR(pin_number)`, where `pin_number` is replaced by a numeric value representing the I/O pin to which the DS18B20 is connected.

In the *Manual*, two different methods are shown for powering the DS18B20: 'normal' and 'parasitic'. We can use either method, but will use the normal method on the *IDM* module. The circuit diagram is shown in Fig.4, and the required modification to the *IDM* shown in Fig.5. The modification is pretty straightforward – comprising the DS18B20 (mounted in the 3-way socket, J8), one resistor, two wire links and two track cuts. Make these modifications, and double check everything as you work.

Note the orientation of the DS18B20 sensor – if you leave the three leads straight, then the flat surface on the sensor should be pointing towards the IR-sensor/piezo-sounder; however, if you're bending the leads at right-angles, then the flat surface needs to be facing up, as shown in Fig.5.

To check everything works, just add a simple test program by inserting the following code before the first line of your existing program. Launch the `EDITor` (`F4`), press the `Home` key twice (to place cursor at the start of the first line of code), and then press the `Enter` key a few times to insert some blank lines at the start of the program. Now type the following four lines of code:

```
CLS
DO
  TEXT 80,40,STR$(TEMPR(2)),CM,1,3
LOOP
```

This program is just to check that the modifications to the *IDM* have been carried out correctly, and also that the DS18B20 temperature sensor is working as expected. `RUN` the code (`F2`) and look at the IPS display. You should see a number that represents the current temperature (in °C). If you see an error reported on the console screen then it is likely to be a spelling mistake – simply check that the first four lines of your program match the above. If you see a high number (like 1000) then you have an issue with the data being sent from the sensor. First check that the sensor orientation is correct, and then check that it is securely inserted into the 3-way socket (J8).

Once you have a realistic temperature value shown on the IPS display, gently 'pinch' the sensor between your finger and thumb. This should result in the number changing as the sensor detects a change in temperature – it will increase as it warms up. After a change of value by a few degrees, let go of the sensor and the temperature should then decrease and eventually settle back at ambient temperature.

The test program is very straightforward. The first line simply clears the IPS display, and then a `DO`/`LOOP` is continually repeated where the temperature is displayed in the middle of the IPS display (*x*=`80`, *y*=`40`, *format*=`CM`), using font 1, with a scale of 3. The `TEMPR(2)` command retrieves the temperature information from the DS18B20 sensor (which is connected to pin 2 on the Micromite) – however, this command returns a number. The `STR$()` command converts this number into a string, or to put it another way, converts the temperature value into a string of alphanumeric characters (which is what the `TEXT` command requires). Please refer to the *User Manual* if you would like more detail regarding the `STR$()` command.

Now that you have added the digital thermometer hardware to your *IDM*, and have tested that it operates correctly, we will incorporate the temperature feature into the *MKC Clock* software. We will make it so that the IR remote can be used to select between date or temperature, shown in the bottom half of the IPS display. In fact, the changes are already incorporated in the program file **MCK_Clock_v3.txt** (referred to above in the *Adding fonts* section). On running v3 of the *MKC Clock* software you will see the same information as before on the IPS display (ie, time and date); however, we now need to configure the four arrow buttons on the IR remote to control the IPS display. This is so the following functionality can be implemented:

`Left arrow`	Display the date (in the bottom part of the screen) ie, underneath the time
`Right arrow`	Display temperature (underneath the time)
`Up arrow`	Increase backlight brightness
`Down arrow`	Decrease backlight brightness

To configure the four arrow buttons, `RUN` the program and then press each arrow button in turn on the IR remote. Make a note of the value displayed on the console screen for each of the four arrow buttons – we will need to know them for the next step. On my remote it is: `LEFT=16`, `RIGHT=90`, `UP=24`, and `DOWN=74`.

Now you need to `EDIT` the code in `SUB IR_Int` to ensure that the four values you noted above are incorporated into the four `CASE` lines of code (the comments should make it clear what you need to do). Once this is done, `RUN` the code and perform the following tests:

1. Press the right arrow button and check that the date is replaced with the temperature information on the IPS display
2. Press the left arrow button and check that the date returns to the IPS display
3. Slowly repeat presses of the down arrow button. Each press results in the backlight getting dimmer (along with a beep). When it reaches its minimum setting, no beep is heard
4. Slowly repeat presses of the up arrow button. Each press results in the backlight getting brighter (along with a beep) until it reaches maximum brightness

If any of the above steps fail, check that the four `CASE` values have been entered correctly. If there is no response at all, then check that the IR receiver is inserted correctly into its socket.

Do take a look at the code and try to understand what is happening. If you are feeling adventurous, you could edit the program slightly so that every five minutes (=300,000ms), the temperature is written to the console screen. This way, you can have a visual log of the temperature over time. To achieve this, insert a new `SETTICK 300000,D_Log` in the `SETUP` section. This will call a new subroutine (that we have called `SUB D_Log`) every five minutes. Between `SUB D_Log` and `END SUB`, simply use a single line of code: `PRINT Date$,Time$,STR$(TEMPR(2))`

Leave this to run for a few hours (or days) and check that the temperature is being recorded to the console screen every five minutes. (The five-minute interval can be adjusted.)

Something else to try

Throughout this series we have demonstrated a number of building blocks that cover a wide range of topics. Why not think about how you could incorporate any combination of what you have learnt into a new project idea of your own. If you need an example to develop further, try building a digital thermometer that has multiple sensors. Think about how to display the multiple temperatures on the IPS display. Possibly use the IR remote to select which sensor's temperature is displayed; alternatively, use the console screen and display a table of temperatures from all the sensors simultaneously (using escape codes to format the console screen). Try adding high and low temperature alarm points (use `INPUT` to get alarm values from the keyboard) and sound various audible alarms on the piezo sounder should any alarm point be exceeded. If anything seems complicated, simply break your project idea down into a series of smaller steps, and then write these steps down on pieces of paper and sequence them appropriately. From there, you can then try writing the code to perform each of the steps involved.

Everything suggested above has been a topic that at some stage has been covered in the series so far. Do try and write your own project in the coming weeks (no matter how big or small), and if you need any help or guidance then simply send me an email.

Next month

In *Part 11*, we will demonstrate how easy it is to connect the Micromite to communicate via the SPI and I^{2}C protocols. We will use an SPI LED matrix module to display alphanumeric characters; and an I^{2}C keypad module with a 4×4 capacitive touch keypad. We'll work through the code that allows communication to and from these modules, enabling us to create a mini project. A finger touch on one of the sixteen numbered pads will display the 'pad number' on the SPI LED matrix; with the ultimate aim of building an electronic combination lock. Until then, have fun!

ETI BUNDLE (1) Teach-In 3, 4 and 5 – all on CD-ROM – only £18.95

ELECTRONICS TEACH-IN 3 – CD-ROM

Mike & Richard Tooley

The three sections of the *Teach-In 3* CD-ROM cover a huge range of subjects that will interest everyone involved in electronics – from newcomers to the hobby and students to experienced constructors and professionals.

The first section (80 pages) is dedicated to *Circuit Surgery*, *EPE/PE's* regular clinic dealing with readers' queries on circuit design problems – from voltage regulation to using SPICE circuit simulation software.

The second section – *Practically Speaking* – covers hands-on aspects of electronics construction. Again, a whole range of subjects, from soldering to avoiding problems with static electricity and identifying components is covered. Finally, our collection of *Ingenuity Unlimited* circuits provides over 40 circuit designs submitted by readers.

The CD-ROM also contains the complete *Electronics Teach-In 1* book, which provides a broad-based introduction to electronics in PDF form, plus interactive quizzes to test your knowledge and TINA circuit simulation software (a limited version – plus a specially written TINA Tutorial).

The *Teach-In 1* series covers everything from electric current through to microprocessors and microcontrollers, and each part includes demonstration circuits to build on breadboards or to simulate on your PC.

ELECTRONICS TEACH-IN 4 – CD-ROM
A BROAD-BASED INTRODUCTION TO ELECTRONICS

Mike & Richard Tooley

The *Teach-In 4* CD-ROM covers three of the most important electronics units that are currently studied in many schools and colleges. These include, Edexcel BTEC level 2 awards and the electronics units of the Diploma in Engineering, Level 2.

The CD-ROM also contains the full *Modern Electronics Manual*, worth £29.95. **The Manual** contains over 800 pages of electronics theory, projects, data, assembly instructions and web links.

A package of exceptional value that will appeal to anyone interested in learning about electronics – hobbyists, students or professionals.

ELECTRONICS TEACH-IN 5 – CD-ROM
JUMP START

Mike & Richard Tooley

15 design and build circuit projects for newcomers or those following courses in school and colleges.

The projects are: ■ Moisture Detector ■ Quiz Machine ■ Battery Voltage Checker ■ Solar-Powered Charger ■ Versatile Theft Alarm ■ Spooky Circuits ■ Frost Alarm ■ Mini Christmas Lights ■ iPod Speaker ■ Logic Probe ■ DC Motor Controller ■ Egg Timer ■ Signal Injector Probe ■ Simple Radio Receiver ■ Temperature Alarm.

PLUS

PIC'n' Mix – starting out with the popular range of PIC microcontrollers and *Practically Speaking* – tips and techniques for project construction.

The CD-ROM also contains:

- ■ Complete *Teach-In 2* book, a practical introduction to PIC microprocessors
- ■ MikroElektronika, Microchip and L-Tek PoScope software.

Three Teach-ins for the great price of £18.95

PLUS you also get the contents of the free CD-ROM from each issue...

... so that's another TWO Teach-Ins and The Full Modern Electronics Manual!

What a Bargain!!

ETI BUNDLE (2) Teach-In 6, 7 and 8 – all on CD-ROM – only £18.95

ELECTRONICS TEACH-IN 6 – CD-ROM
A COMPREHENSIVE GUIDE TO RASPBERRY Pi

Mike & Richard Tooley

Teach-In 6 contains an exciting series of articles that provides a complete introduction to the Raspberry Pi, the low cost computer that has taken the education and computing world by storm.

This latest book in our *Teach-In* series will appeal to electronics enthusiasts and computer buffs who want to get to grips with the Raspberry Pi.

Teach-In 6 is for anyone searching for ideas to use their Pi, or who has an idea for a project but doesn't know how to turn it into reality. This book will prove invaluable for anyone fascinated by the revolutionary Pi. It covers:

- ■ Pi programming
- ■ Pi hardware
- ■ Pi communications
- ■ Pi Projects
- ■ Pi Class
- ■ Python Quickstart
- ■ Pi World
- ■ ...and much more!

The *Teach-In 6* CD-ROM also contains all the necessary software for the series, so that readers and circuit designers can get started quickly and easily with the projects and ideas covered.

ELECTRONICS TEACH-IN 7 – CD-ROM
DISCRETE LINEAR CIRCUIT DESIGN

Mike & Richard Tooley

Teach-In 7 is a complete introduction to the design of analogue electronic circuits. It is ideal for everyone interested in electronics as a hobby and for those studying technology at schools and colleges. The CD-ROM also contains all the circuit software for the course, plus demo CAD software for use with the *Teach-In* series.

- ■ Discrete Linear Circuit Design
- ■ Understand linear circuit design
- ■ Learn with 'TINA' – modern CAD software
- ■ Design simple, but elegant circuits
- ■ Five projects to build:
  - i) Pre-amp
  - ii) Headphone Amp
  - iii) Tone Control
  - iv) VU-meter
  - v) High Performance Audio Power Amp.

PLUS

Audio Out – an analogue expert's take on specialist circuits

Practically Speaking – the techniques of project building.

ELECTRONICS TEACH-IN 8 – CD-ROM
INTRODUCING THE ARDUINO

Mike & Richard Tooley

Hardware: learn about components and circuits
Programming: powerful integrated development system
Microcontrollers: understand control operations
Communications: connect to PCs and other Arduinos.

Teach-In 8 is an exciting series designed for electronics enthusiasts who want to get to grips with the inexpensive, popular Arduino microcontroller, as well as coding enthusiasts who want to explore hardware and interfacing. It will provide a one-stop source of ideas and practical information.

The Arduino offers a truly effective platform for developing a huge variety of projects; from operating a set of Christmas tree lights to remotely controlling a robotic vehicle through wireless or the Internet.

Teach-In 8 is based around a series of practical projects with plenty of information for customisation.

This book also includes *PIC n' Mix*: 'PICs and the PICkit 3 – A Beginners guide' by Mike O'Keefe and *Circuit Surgery* by Ian Bell – 'State Machines part 1 and 2'.

The CD-ROM includes the files for:

- ■ *Teach-In 8*
- ■ Microchip MPLAB IDE XC8 8-bit compiler
- ■ PICkit 3 User Guide
- ■ Lab-Nation Smartscope software.

ORDER YOUR BUNDLE TODAY!
JUST CALL 01202 880299 – OR VISIT www.epemag.com

Max's Cool Beans

By Max the Magnificent

The real work begins!

Way back in the mists of time (*PE*, February 2019), I introduced my *Countdown Timer* project, whose main mission in life is to display the years (YY), months (MM), days (DD), hours (HH), minutes (MM), and seconds (SS) to my 100th birthday. The festivities will commence at 11:45am British Summer Time on 29 May 2057, so now would be a good time to clear this date on your calendar.

In the aforementioned *Cool Beans* column, we considered a couple of interesting special effects – using a cross-fade when transitioning from one digit to another in the normal course of events, and a scroll-back when transitioning from 9 to 0 (when counting up) or 0 to 9 (when counting down).

More recently (*PE*, August 2019), I showed pictures of my friend, master carpenter Bob, along with the *Countdown Timer's* cabinet in the process of being constructed. Shortly after these pictures were taken, we completed and stained the cabinet and I added the Lixie displays and brass accoutrements (Fig.1).

The next step was to add a processor in the form of a Teensy 3.6 from PJRC (**https://bit.ly/2JZ94Fw**), an ultra-precise real-time clock (RTC) in the form of a ChronoDot v2.1 from Adafruit (**https://bit.ly/2m6p7Gc**), and three tactile switches along with an LS118 3-channel switch debounce IC from LogiSwitch (**https://bit.ly/2GnufxA**).

I then spent a happy day wiring everything together (Fig.2.) So, that's the easy stuff over; now the real (programming) work begins. Eeek!

Data, data everywhere...

Now, I love watching countdown timers doing their thing as much as the next geek. 'Show me a flashing LED, and I'll show you a man drooling,' as I often say. But I suspect that even I might begin to get bored watching a timer perform a simple countdown after the first 10 years or so.

Thus, the next thing I decided to do is alternate between displaying the time remaining to my 100th birthday and the amount of time I've already graced this planet with my presence, both values taking things like leap years into account, and both being presented in YY, MM, DD, HH, MM, SS format. But then I thought that it might be interesting to also present these values in the form of the total number of days (DD, DD, DD, DD, DD, DD), then the total number of hours (HH, HH, HH, HH, HH, HH), then the total number of minutes (MM, MM, MM, MM, MM, MM), and finally the total number of seconds (SS, SS, SS, SS, SS, SS).

Since the maximum value of 999,999,999,999 seconds equates to approximately 31,710 years, I think we have sufficient capacity to hand, even taking potential future life-extending medical advances into account.

Next, it struck me that it would be silly to have an artifact like the *Countdown Timer* and not be able to use it to tell the time, so I decided that – since I presently hang my hat in Huntsville, Alabama – the timer should periodically display the current time in US Central Time and the corresponding time back 'home' in Yorkshire, taking things like Daylight Saving Time (DST) and British Summer Time (BST) into account, of course.

As an aside, did you know that – although we 'jump forward' and 'fall back' on different days – both the Americans and the British regard wintertime as being the standard, with DST or BST being a positive offset. By comparison, some countries like Ireland take the view that summertime is the standard, with wintertime being a negative offset.

But we digress... the point is that we are now going to be using the *Countdown Timer* to display so many different things that we're going to require a secondary display to tell us what we're looking at!

Now, we could use something like a cheap-and-cheerful LCD character display, such as a 20-character by 4-row offering from Adafruit (**https://bit.ly/2kAM97z**), but this really wouldn't be in keeping with the look-and-feel of the main system. What I would really like is a small CRT-based (cathode ray tube) display mounted in a wooden case that matches the *Countdown Timer*, but how would we drive such a display?

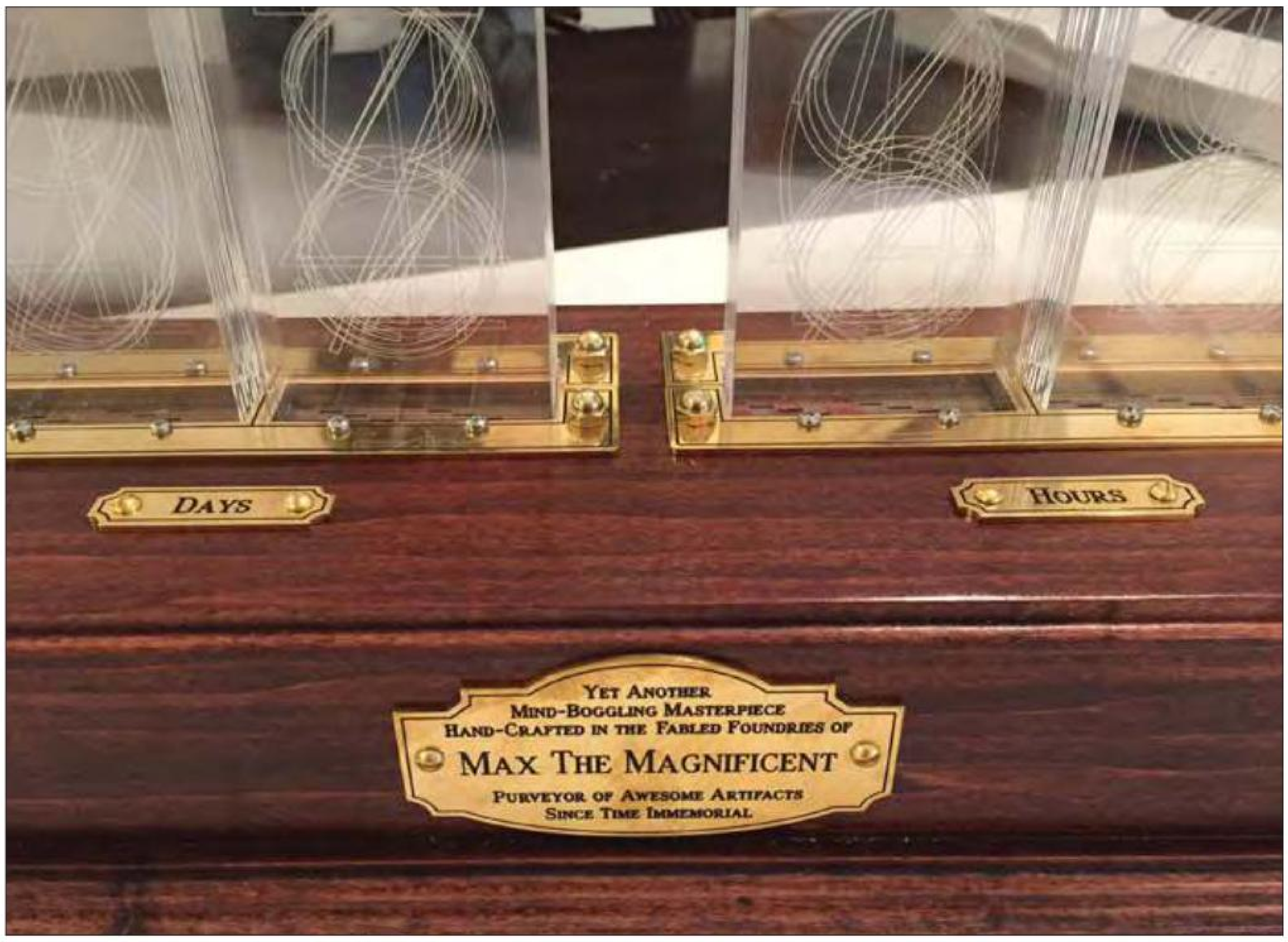
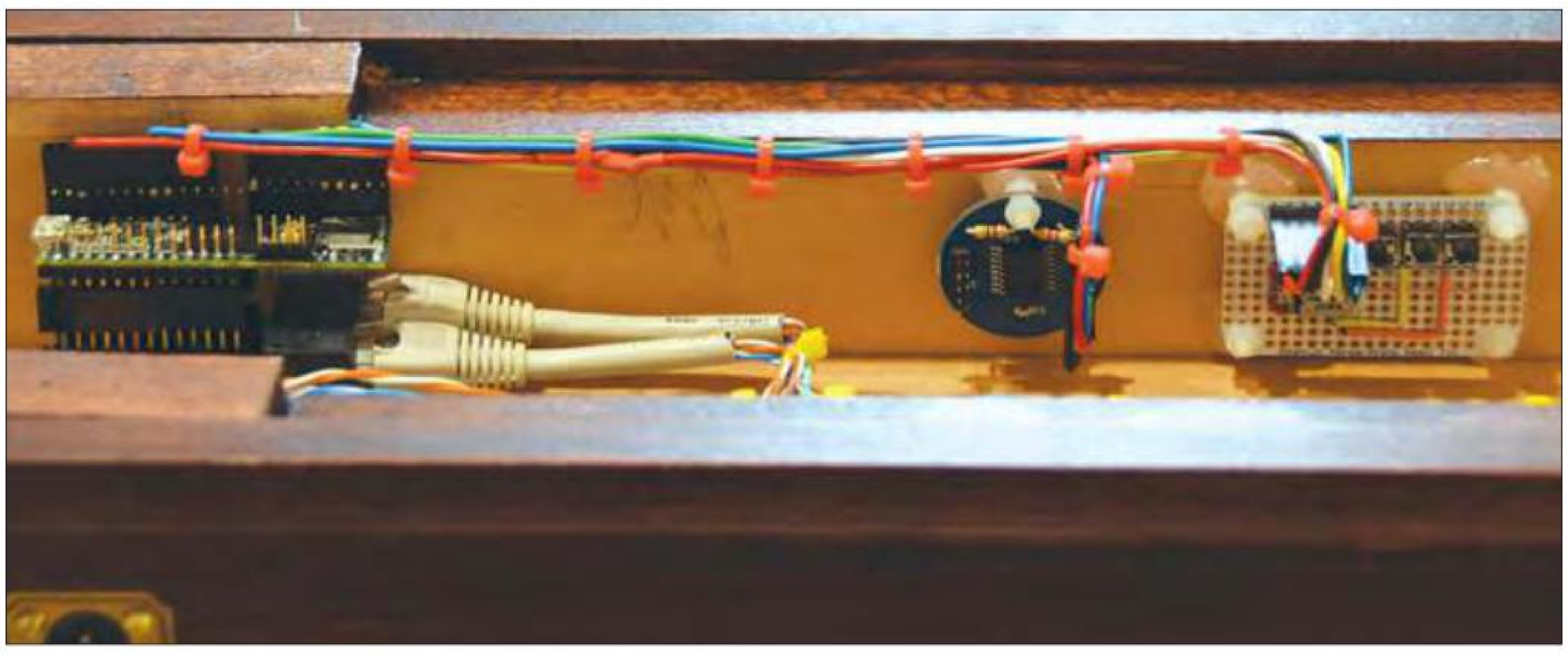

DAYS
HOURS
YET ANOTHER
MIND-BOGGLING MASTERPIECE
HAND-CRAFTED IN THE FABLED FOUNDRIES OF
MAX THE MAGNIFICENT
PURVEYOR OF AWESOME ARTIFACTS
SINCE TIME IMMEMORIAL

Fig.1. Adding the *Countdown Timer's* Lixie displays and brass accoutrements.

VT100s rule!

In the early days of home computing, memory tended to be limited in size and expensive. A common technique was to use a memory-mapped display, in which a special area of the computer's system RAM was set aside to be the 'screen memory.' If we think of the screen as being divided into an array of rows and columns, where each cell can contain a single ASCII character (alpha,



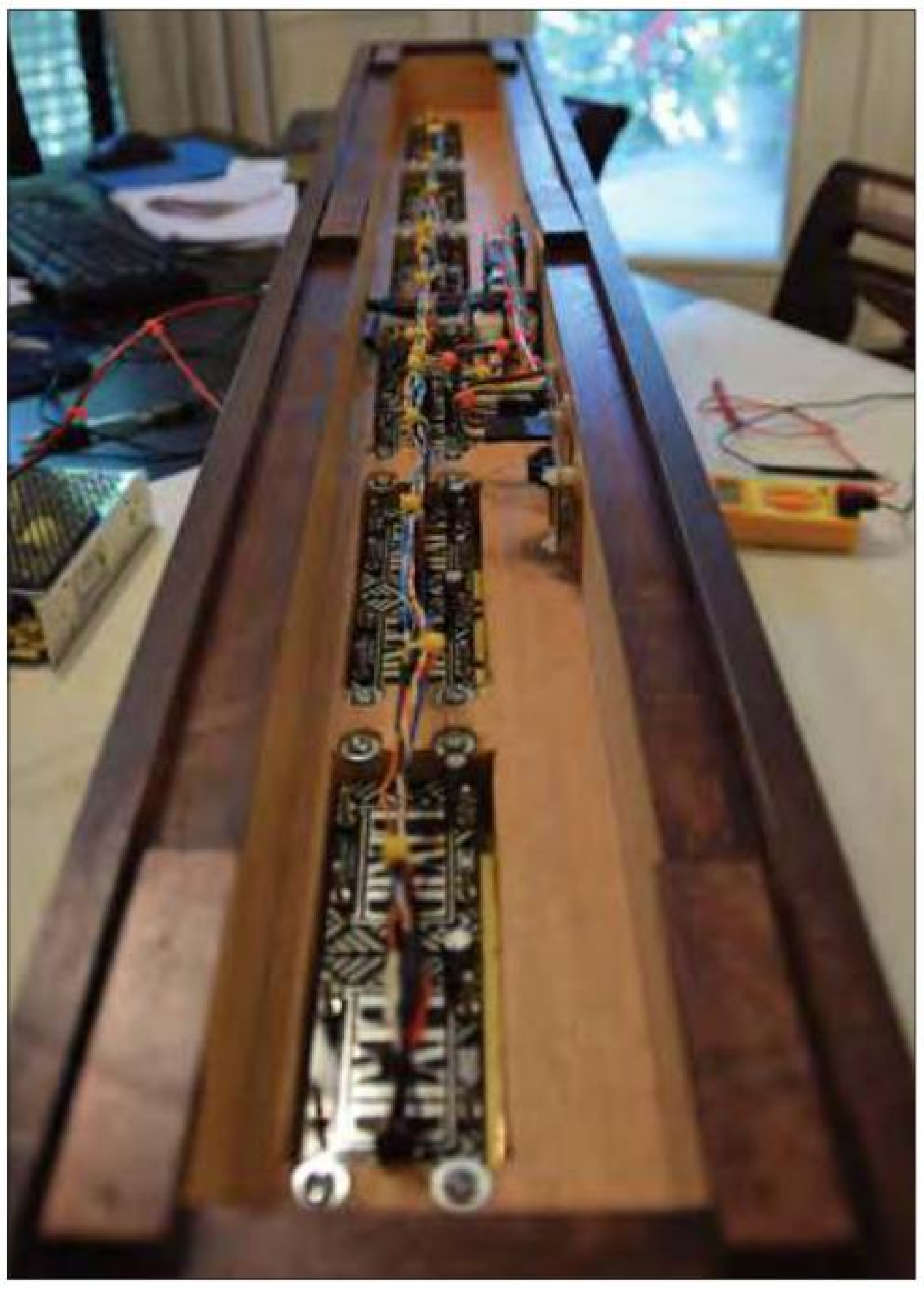


Fig.2. (above and right) Wiring everything together.

numeric, and punctuation), then each byte in the screen memory would hold an ASCII character.

The CPU would write ASCII characters into these screen memory locations, and a separate circuit would read them out of the screen memory and implement all the slicing, dicing, and timing required to present them on the raster-scanned CRT screen.

Sadly, the memory-mapped display technique is of no use to us here, however… a rather interesting development occurred in 1978, when Digital Equipment Corporation (DEC) introduced their VT100 video terminal (Fig.3). The VT100 contained its own screen memory, along with an 8-bit Intel 8080 microprocessor.

Of particular interest was the fact that the VT100 supported a suite of ANSI escape sequences for cursor control and other tasks. Each sequence comprised a series of ASCII characters, commencing with the 'Escape' character (code 27 in decimal or 0x1B in hexadecimal). These sequences allowed you to do things like specify the *x,y* location of the cursor on the screen, turn reverse video on or off, and make the cursor a flashing underline or a flashing block.

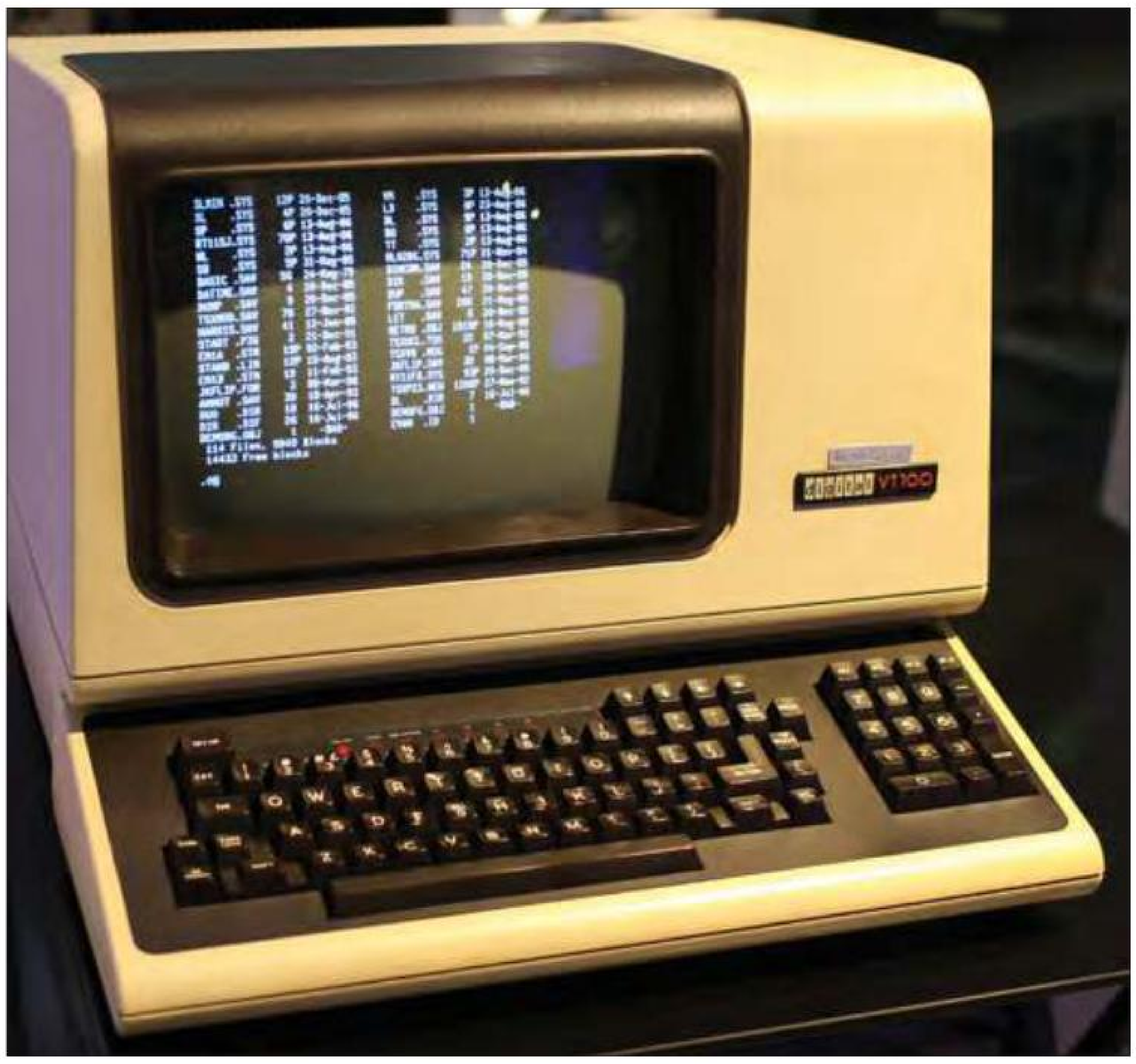


Fig.3. DEC VT100 terminal (Source: Jason Scott/Wikipedia).

If I happened to have a VT100 terminal in my collection, this would be perfect for what I'm trying to do. The Teensy 3.6 in my *Countdown Timer* could use one of its five UART interfaces to write ANSI escape sequences and ASCII text to the terminal, which would handle the actual display tasks. Furthermore, I could use the terminal's keyboard to transmit commands to the *Countdown Timer*, like changing the time or testing a new display mode.

I'm so excited with this idea (happy face). The only flaw with this cunning plan is that I don't actually own a VT100 (sad face).

VT100 Emulator

While I was bemoaning my lack of a VT100 and mulling things over, I ran across a project by Geoff Graham in the July 2014 issue of *Silicon Chip* magazine. This little beauty (the project, not Geoff) is a complete ASCII VT100-compatible terminal implemented on a small circuit board that basically comprises a PIC microcontroller from Microchip Technology, a handful of discrete components, and a bunch of connectors (**https://bit.ly/2kzSOPw**).

This device accepts a serial input stream of ANSI escape sequences and ASCII text characters from a UART interface on the microcontroller of your choice (Arduino, Teensy, Raspberry Pi, Micromite etc.), and it generates all the necessary timing to output signals to drive any CRT or LCD display that supports VGA (Vector Graphics Array) or composite video inputs.

'Could things get any better?' I hear you cry. Well, yes, they could, actually, because I emailed my chum, Joe Farr, in the UK to tell him about this little scamp. Joe is my co-conspirator on the *4-bit HRRG* mixed technology computer we are building, and he creates lots of hobby projects that could use Geoff's VT100 emulator.

You can only imagine my surprise a couple of days later when I received an email from Joe saying he had redesigned Geoff's board. It seems that Joe couldn't get Geoff's Gerber files to load correctly in his PCB package, so he decided to create his own layout.

Apart from anything else, this gave Joe the opportunity to change various component footprints to suit what he already had or what he could easily get hold of. In particular, the original video connector was hard to source, so Joe replaced it with a later version.

Joe also made a couple of other changes, such as adding a DC barrel connector and a bridge rectifier with smoothing capacitors and a 5V regulator. As a result, Joe's version can be powered by a 7V to 30V AC or DC source (a heatsink will be required for the 5V regulator if the input voltage exceeds around 7V). Knowing that I prefer to use a regulated 5V supply in my projects, Joe also provided a header that allows me to push 5V straight in.

A few days later, I found a small package in my post box. When I opened it, I discovered a fully assembled version of Joe's board (Fig.4).

On the off-chance you want to make one of these little scamps for yourself, Joe has very kindly provided the full Gerber file set that he used to build this board (**https://bit.ly/2m3F0wZ**); meanwhile, you can download the PIC firmware from Geoff's website as noted earlier, or you can purchase a pre-programmed PIC from *Silicon Chip's* website (**https://bit.ly/2m5iNyH**).

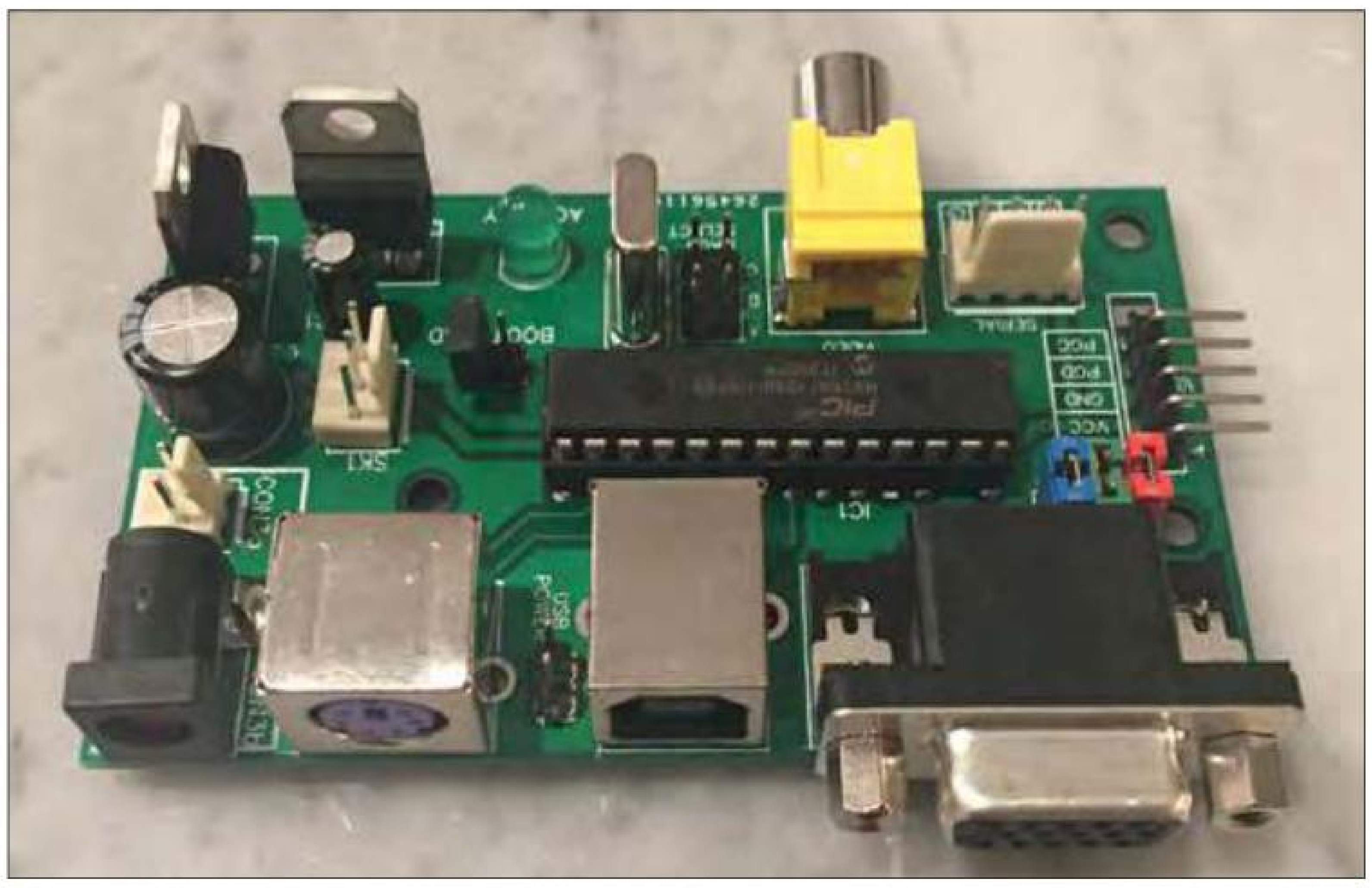


Fig.4. Joe Farr's reinterpretation of Geoff Graham's VT100 emulator.

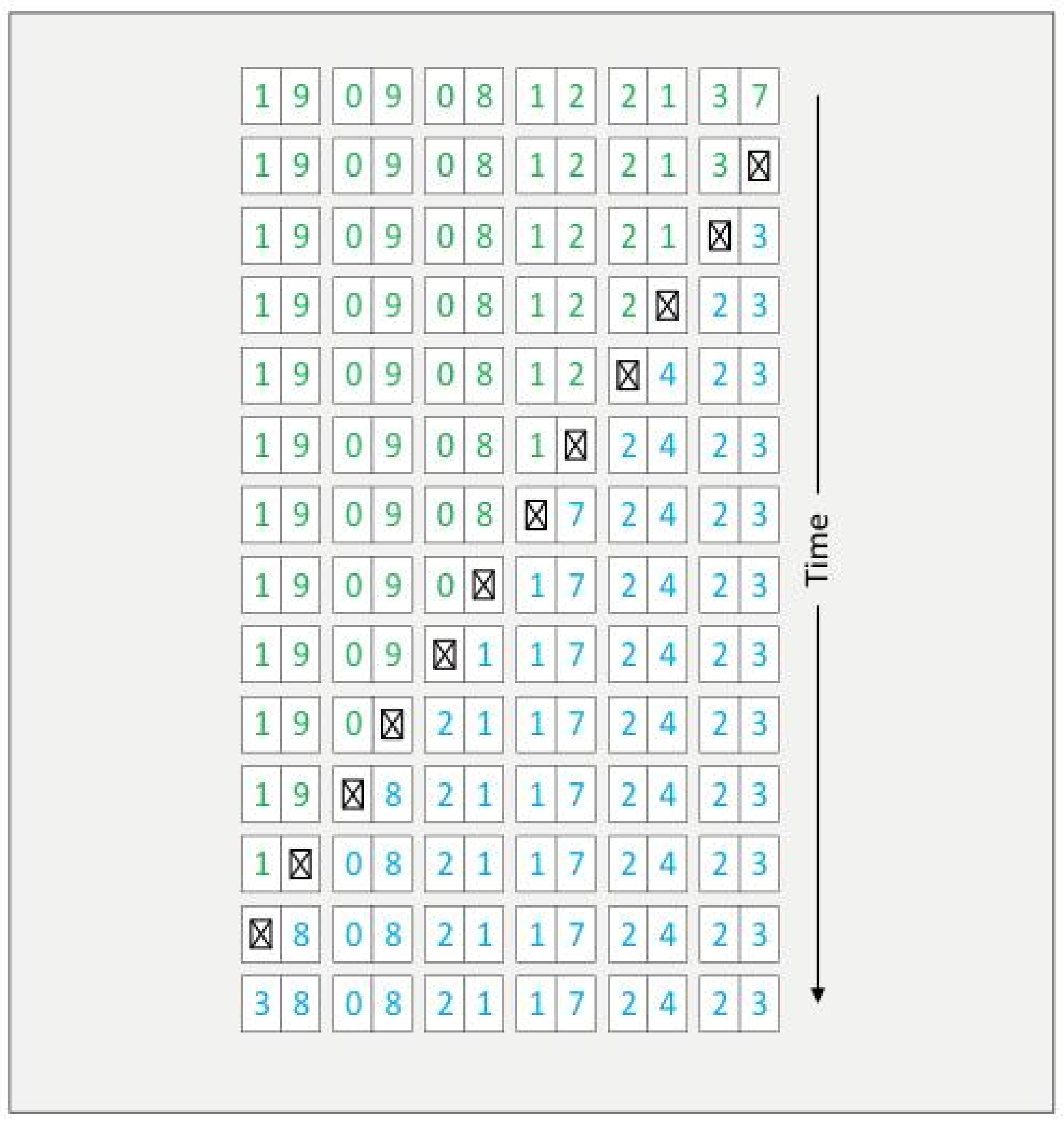
In transition

As a tempting teaser, I've uploaded a video to YouTube showing the *Countdown Timer* using its real-time clock to display the current time (**https://youtu.be/Cg83IxobFT8**). As you will see, this video commences by displaying 19 08 21 19 54 46. Just to make sure we're all tap-dancing to the same drumbeat, the 19 on the left represents the year 2019, while the 19 in the middle represents the hour in 24-hour format. The first thing I'm going to do is change this to display the hour in 12-hour format because I think this will make more sense to the average observer.

OK, so once I find a suitable display for my VT100 emulator, I'll know what information is currently being presented on the main display; but how are we going to transition from one value to another? For example, let's say that the current date and time as I pen these words is 19 09 08 12 21 37, and I want to transition from this to display how long I have to go to the commencement of my 100th birthday celebrations. I'm still trying to wrap my brain around how to calculate this, but – just for the sake of these discussions – let's say that we have 38 08 21 17 24 23 to go.

So, how are we going to transition from 19 09 08 12 21 37 to 38 08 21 17 24 23? We could simply swap over from one value to another, but if we did this while we were blinking, we might not even notice the change. One option would be to display the first value in one color, say green, and the second value in another color, for example blue, but this still isn't tremendously exciting.

What I'm thinking of is a sort of 'roll-over' effect, where the least-significant digit (LSD) starts flicking between different values, then this first digit settles on its new value and color while the next digit starts flicking between different values, and so on up the line to the most-significant digit (MSD). We can visualise this as looking something like the illustration in Fig.6.



1	9	0	9	0	8	1	2	2	1	3	7
1	9	0	9	0	8	1	2	2	1	3	☒
1	9	0	9	0	8	1	2	2	1	☒	3
1	9	0	9	0	8	1	2	2	☒	2	3
1	9	0	9	0	8	1	2	☒	4	2	3
1	9	0	9	0	8	1	☒	2	4	2	3
1	9	0	9	0	8	☒	7	2	4	2	3
1	9	0	9	0	☒	1	7	2	4	2	3
1	9	0	9	☒	1	1	7	2	4	2	3
1	9	0	☒	2	1	1	7	2	4	2	3
1	9	☒	8	2	1	1	7	2	4	2	3
1	☒	0	8	2	1	1	7	2	4	2	3
☒	8	0	8	2	1	1	7	2	4	2	3
3	8	0	8	2	1	1	7	2	4	2	3

Fig.6. A simple roll-over transition from one value to another.

As a starting point, let's say that the transitioning digit will flicker between *n* × 10 values, where *n* could be 1, 2, 3… This transitioning digit could be in a different color, say white, or it could gradually fade from the original color (green in this example) to the new color (blue in this example) over the course of our *n* × 10 flicker values.

Furthermore, we might wish to perform the transition from the MSD to the LSD. Also, in addition to transitioning from one set of numbers to another, we might wish to transition from a set of numbers to a null value (ie, everything turned off), or from a null value to a set of numbers.

With regard to the flickering values themselves, we might wish these to be random numbers, or we might wish them to be a count sequence. In this latter case, we could have them counting up from 0 to 9 in the case of an LSD-to-MSD transition, or counting down from 9 to 0 in the case of an MSD-to-LSD transition.

In Fig.6, we showed only a single flickering digit, but why couldn't we have a block of two, three, or more? In this case, assuming an LSD-to-MSD transition, and also assuming a block of three flickering digits, the LSD would start flickering first, then the digit next to it, then the digit next to that, at which point the entire block of three flickering digits would make its way up the display (Fig.7).

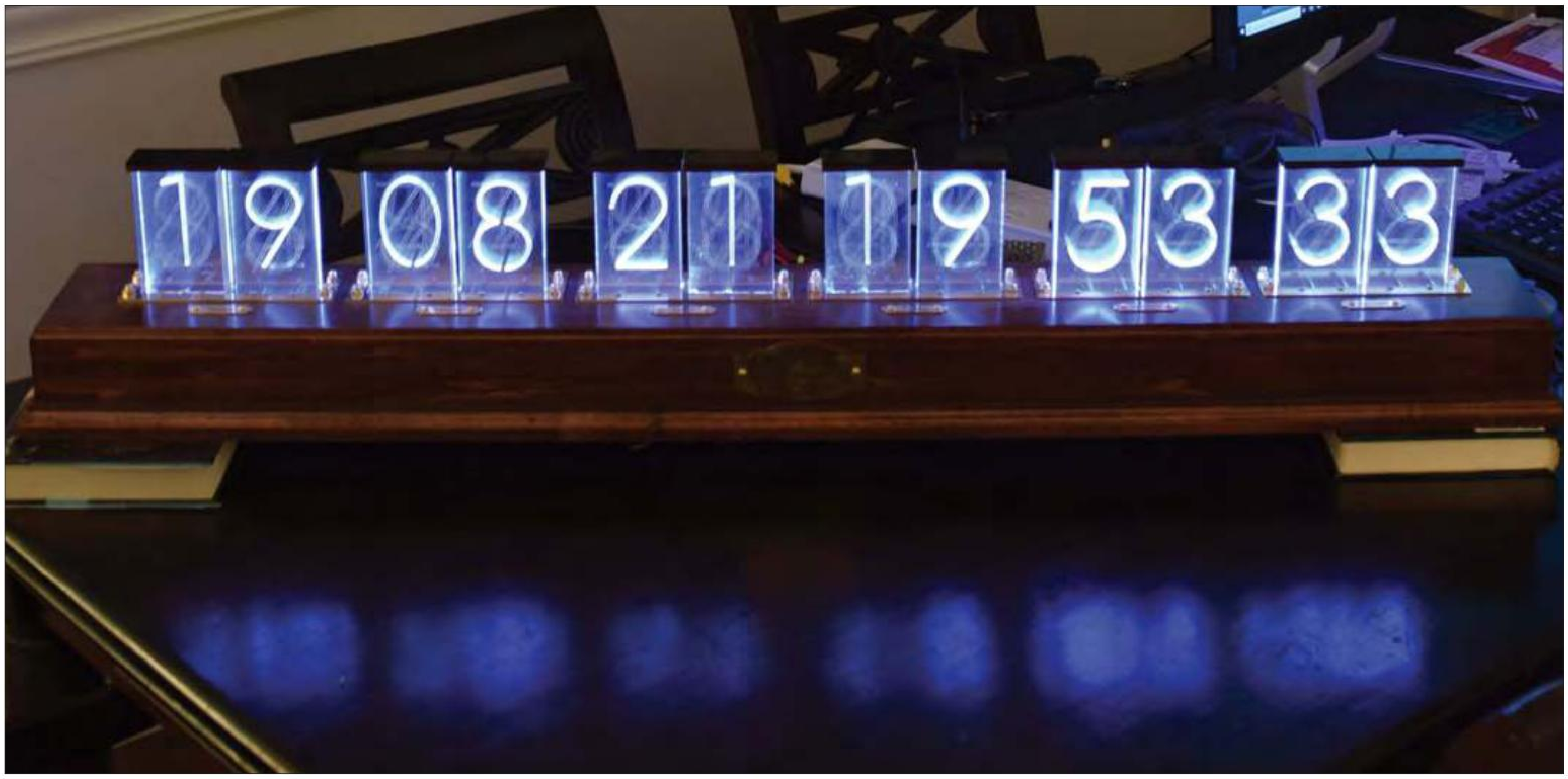


Fig.5. A first view of the *Countdown Timer* displaying the current time.

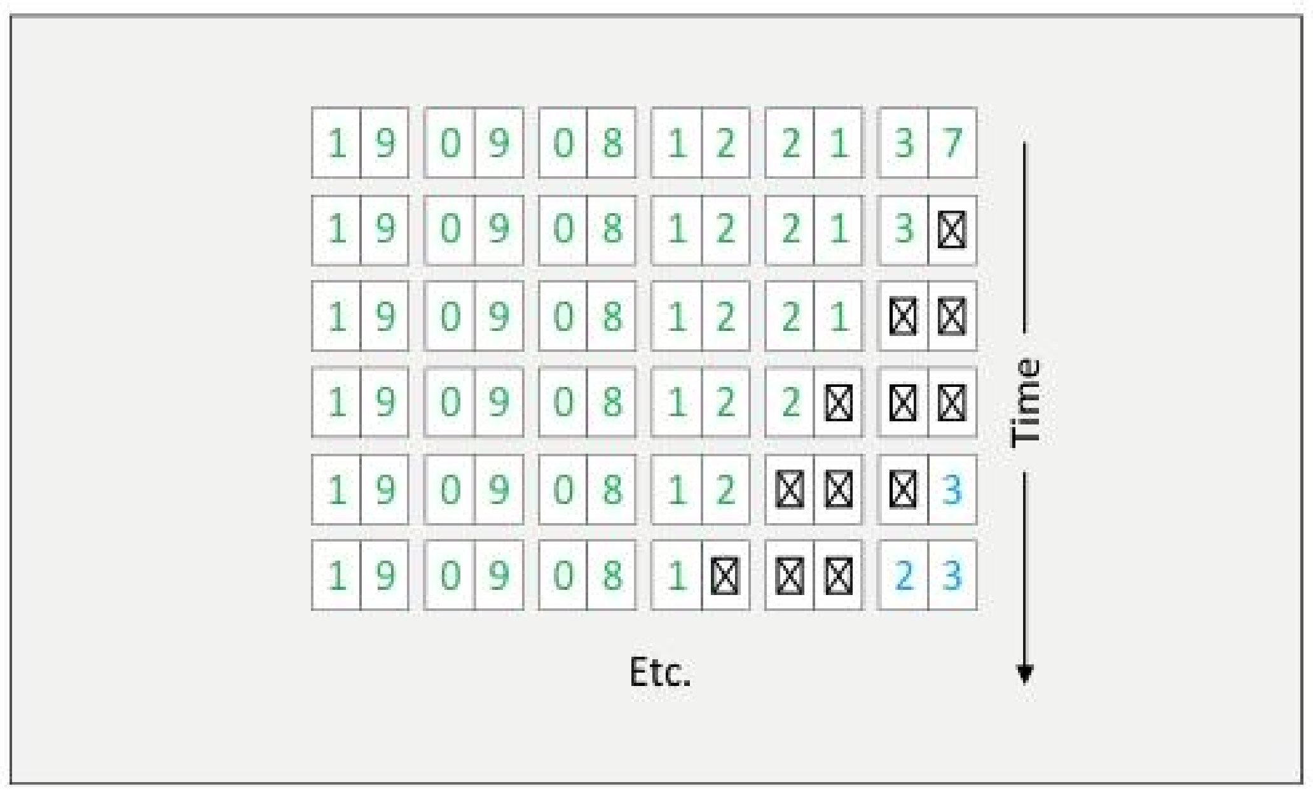


Fig.7. A more-sophisticated roll-over transition from one value to another.

The problem with this sort of thing is that you don't know which effect you like best until you see them all in action. Also, you might like to use different versions of the effects in different situations. Since my Teensy 3.6 microcontroller has computing power to spare, the way I've addressed this is to create a single, highly parameterised function that can implement all of the aforementioned effects – you can select the start and end colors, a special transition color or a fade between the start and end colors, LSD-to-MSD or MSD-to-LSD transitions, the number of cycles (*n*) and the number of digits flickering at any particular time, and so forth.

You can see some examples of this in another video I just uploaded to YouTube (**https://bit.ly/2lZd0KT**). Now, remembering that I'm a hardware design engineer by trade and that my programming skills leave something to be desired, I'm more than happy for you to take a look at my code for this roll-over transition effect and play with it as you wish (**https://bit.ly/2kA1mFW**).

The final countdown

Last but not least, for this column, we need to consider what we should do when we reach the final countdown, which will occur at 11:45am BST (5:45am DST) on 29 May 2057.

I think it goes without saying that the 00 00 00 00 00 00 value will start flashing on and off, and it will continue to do so throughout the day. I'm also envisaging some cheerful music that will become more annoying as the day progresses. If I'm around to see this, then I will have a way to turn the music off. However, if I'm otherwise engaged (canoeing up the mighty Amazon river, for example), then I can envisage my son disconnecting the power after a couple of hours. I think this would be perfectly understandable, but I also think I will include backup battery so that the music will start up again after an hour or so when he's least expecting it.

Returning to the scenario where I am present to enjoy the occasion, I'm envisioning the flashing 0s to persist until 12:00 midnight of that day, but what should we do after that? My initial thought was for the counter to reset itself to commence a countdown to my 110th birthday, and then my 120th, and so forth, but this is just 'more of the same.'

As an alternative, I think it's better to start displaying the amount of additional time (YY MM DD HH MM SS) I've been granted following my first 100 years, on the basis that each additional second is a gift.

So, that's where things stand at the moment. I will continue to happily program away (I just hope I finish the programming before the counter reaches 0). In the meantime, as always, I look forward to receiving your comments, questions, and suggestions. Until next time, have a good one!

Cool bean Max Maxfield (Hawaiian shirt, on the right) is emperor of all he surveys at **CliveMaxfield.com** – the go-to site for the latest and greatest in technology.

Comments or questions? Email Max at: **max@CliveMaxfield.com**

FREE CD-ROM
ELECTRONICS TEACH-IN 9
GET TESTING!
PIC n' Mix
Teach-In 9 Get Testing

GET THE LATEST COPY OF OUR TEACH-IN SERIES AVAILABLE NOW!

ON SALE in WHSmith and other independent newsagents

PRICE £8.99
(includes P&P to UK if ordered direct from us)

Teach-In 9 – Get Testing!

This series of articles provides a broad-based introduction to choosing and using a wide range of test gear, how to get the best out of each item and the pitfalls to avoid. It provides hints and tips on using, and – just as importantly – interpreting the results that you get. The series deals with familiar test gear as well as equipment designed for more specialised applications.

The articles have been designed to have the broadest possible appeal and are applicable to all branches of electronics. The series crosses the boundaries of analogue and digital electronics with applications that span the full range of electronics – from a single-stage transistor amplifier to the most sophisticated microcontroller system. There really is something for everyone!

Each part includes a simple but useful practical test gear project that will build into a handy gadget that will either extend the features, ranges and usability of an existing item of test equipment or that will serve as a stand-alone instrument. We've kept the cost of these projects as low as possible, and most of them can be built for less than £10 (including components, enclosure and circuit board).

FREE COVER-MOUNTED CD-ROM

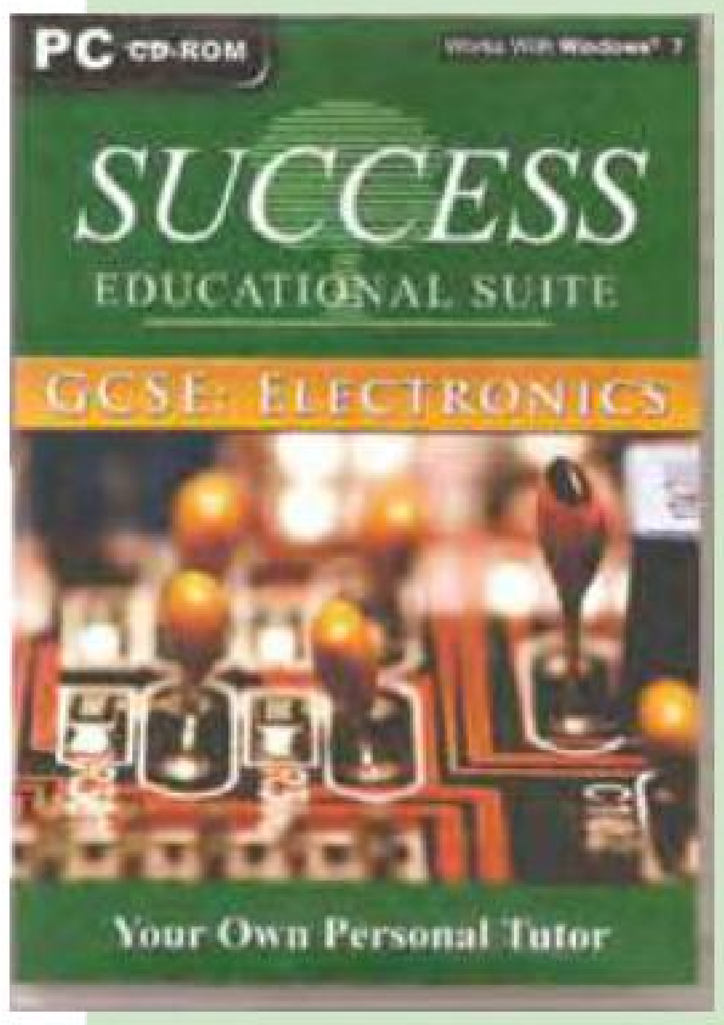
On the free cover-mounted CD-ROM you will find the software for the *PIC n' Mix* series of articles. Plus the full *Teach-In 2* book – *Using PIC Microcontrollers – A practical introduction* – in PDF format. Also included are Microchip's *MPLAB ICD 4 In-Circuit Debugger User's Guide*; *MPLAB PICkit 4 In-Circuit Debugger Quick Start Guide*; and *MPLAB PICkit 4 Debugger User's Guide*.

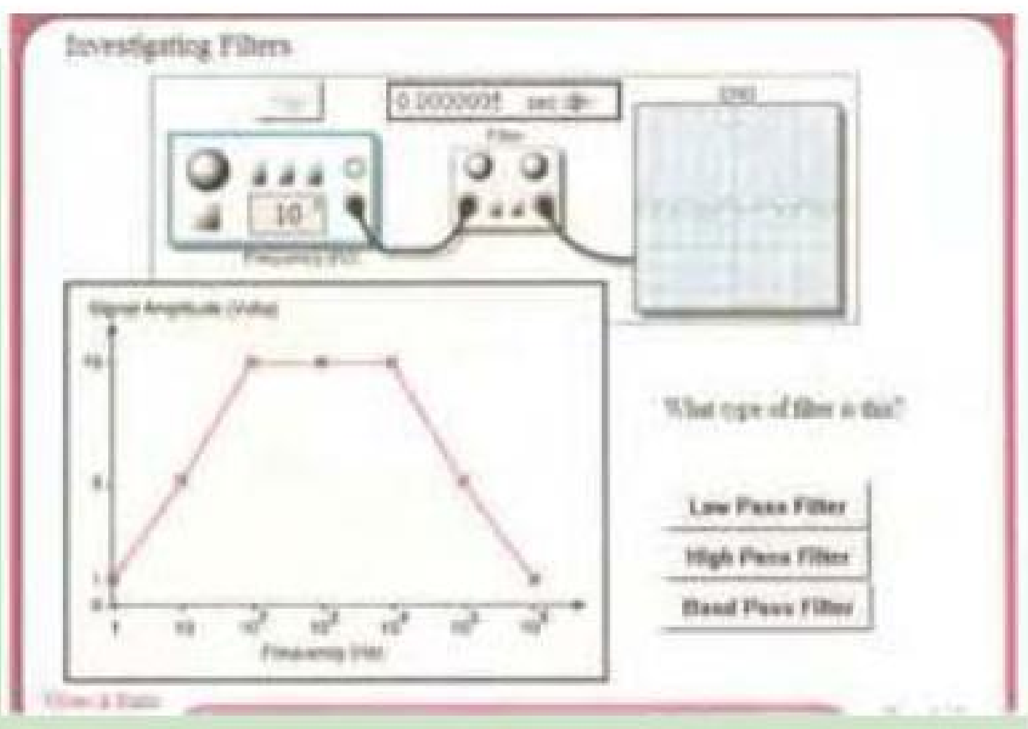
ORDER YOUR COPY TODAY JUST CALL
01202 880299 OR VISIT www.epemag.com

PRACTICAL ELECTRONICS IS PLEASED TO OFFER YOU THESE

ELECTRONICS CD-ROMS

GCSE ELECTRONICS





Suitable for any student who is serious about studying and who wants to achieve the best grade possible. Each program's clear, patient and structured delivery will aid understanding of electronics and assist in developing a confident approach to answering GCSE questions. The CD-ROM will be invaluable to anyone studying electronics, not just GCSE students.

*** Contains comprehensive teaching material to cover the National Curriculum syllabus * Regular exercises reinforce the teaching points * Retains student interest with high quality animation and graphics * Stimulates learning through interactive exercises * Provides sample examination ques-tions with model solutions * Authored by practising teachers * Covers all UK examination board syllabuses * Caters for all levels of ability * Useful for self-tuition and revision**

SUBJECTS COVERED

Electric Circuits – Logic Gates – Capacitors & Inductors – Relays – Transistors – Electric Transducers – Operational Amplifiers – Radio Circuits – Test Instruments

Over 100 different sections under the above headings

£12.50
inc. VAT and P&P

CIRCUIT WIZARD

Circuit Wizard is a revolutionary software system that combines circuit design, PCB design, simulation and CAD/CAM manufacture in one complete package. Two versions are available – Standard or Professional.

By integrating the entire design process, Circuit Wizard provides you with all the tools necessary to produce an electronics project from start to finish – even including on-screen testing of the PCB prior to construction!

* Circuit diagram design with component library (500 components Standard,1500 components Professional) * Virtual instruments (4 Standard, 7 professional) * On-screen animation * Interactive circuit diagram simulation * True analogue/digital simulation * Simulation of component destruction * PCB Layout * Interactive PCB layout simulation * Automatic PCB routing * Gerber export * Multi-level zoom (25% to 1000%) * Multiple undo and redo * Copy and paste to other software * Multiple document support

This software can be used with the *Jump Start* and *Teach-In 2011* series (and the *Teach-In 4* book).

Standard **£61.25** inc. VAT. Professional **£75** plus VAT.

From £49.00

TINA Design Suite V12

Analogue, Digital, Symbolic, RF, MCU and Mixed-Mode Circuit Simulation and PCB Design with TINA

FEATURED IN OUR TEACH-IN 2015 SERIES

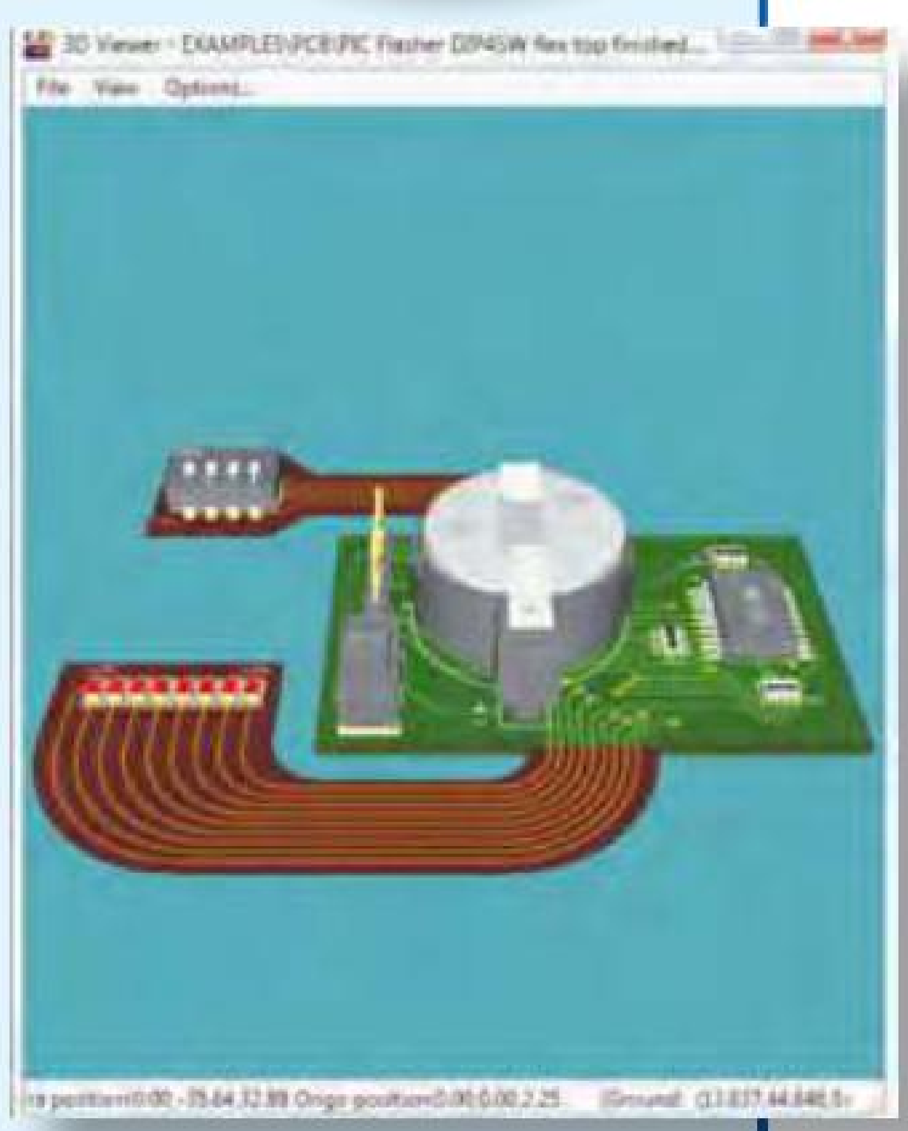
TINA Design Suite V12 is a powerful yet affordable software package for analysing, designing and real time testing analogue, digital, MCU, and mixed electronic circuits and their PCB layouts. You can also analyse RF, communication, optoelectronic circuits, test and debug microcontroller applications.

Enter and analyse any circuit up to 100 nodes (student), or 200 with the Basic (Hobbyist) version within minutes with TINA's easy-to-use schematic editor. Enhance your schematics by adding text and graphics. Choose components from the large library containing more than 10,000 manufacturer models. Analyse your circuit through more than 20 different analysis modes or with 10 high tech virtual instruments.

Present your results in TINA's sophisticated diagram windows, on virtual instruments, or in the live interactive mode where you can even edit your circuit during operation.

Customise presentations using TINA's advanced drawing tools to control text, fonts, axes, line width, colour and layout. You can create and print documents directly inside TINA or cut and paste your results into your favourite word procesing or DTP package.

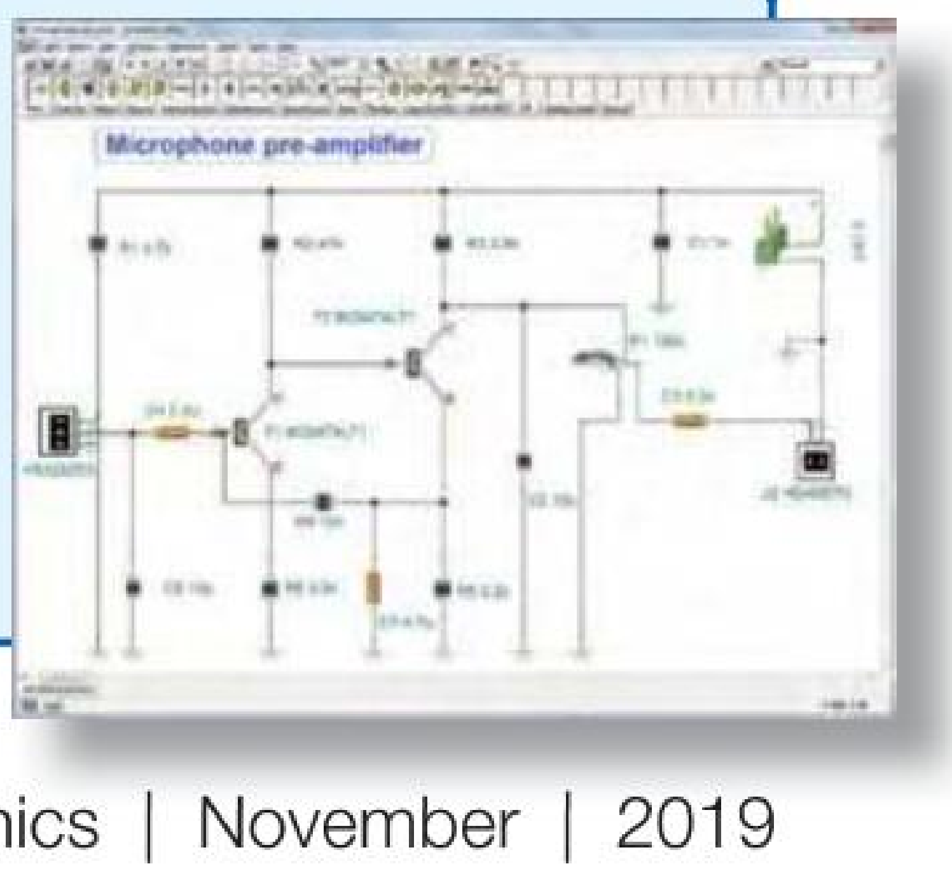
TINA includes the following Virtual Instruments: Oscilloscope, Function Generator, Multimeter, Signal Analyser/Bode Plotter, Network Analyser, Spectrum Analyser, Logic Analyser, Digital Signal Generator, XY Recorder.



This offer gives you a CD-ROM – the software will need registering (FREE) with Designsoft (TINA), details are given within the package.

Get TINA 12 Design Suite (Hobbyist) for £129 or Student V12 version for £49

Prices include VAT and UK postage



+ get a 1 year free subscription for TINACloud the breakthrough cloud version of TINA which you can run on most operating systems and computers, including PCs, Macs, thin clients iPads and other tablets – even on many smart phones, smart TVs and e-book readers.

To order please either fill out and return the order form, or call us on 01202 880299

Alternatively you can order via our secure online shop at: www.epemag.com

PICmicro TUTORIALS AND PROGRAMMING

HARDWARE

PICmicro Multiprogrammer Board and Development Board

Suitable for use with the three software packages listed below

This flexible PICmicro microcontroller programmer board and combination board allows students and professional engineers to learn how to program PICmicro microcontrollers as well as program a range of 8, 18, 28 and 40 pin devices from the 12, 16 and 18 series PICmicro ranges. For those who want to learn, choose one or all of the packages below to use with the hardware.

- Makes it easier to develop PICmicro projects
- Supports low cost Flash-programmable PICmicro devices
- Fully featured integrated displays – 16 individual LEDs, quad 7-segment display and alphanumeric LCD display
- Supports PICmicro microcontrollers with A/D converters
- Fully protected expansion bus for project work
- USB programmable
- Compatible with the E-blocks range of accessories

£159.99 including VAT and postage (worldwide)

SOFTWARE

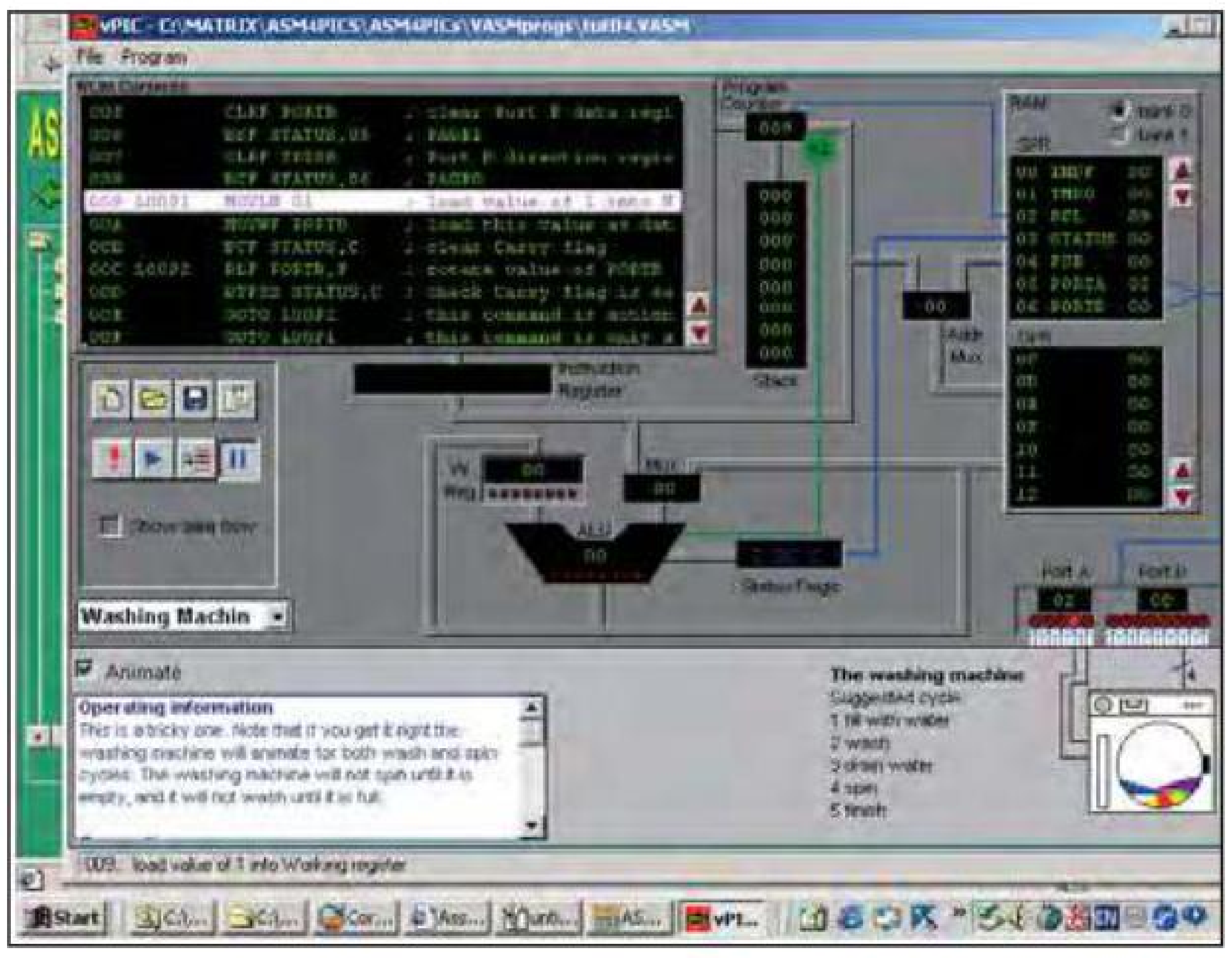
ASSEMBLY FOR PICmicro V7

(Formerly PICtutor)

Assembly for PICmicro microcontrollers V7.0 (previously known as PICtutor) by John Becker contains a complete course in programming the PIC16F84, 16F88 and 16F877a PICmicro microcontroller from Arizona Microchip. It starts with fundamental concepts and extends up to complex programs including watchdog timers, interrupts and sleep modes.

The CD makes use of the latest simulation techniques which provide a superb tool for learning: the Virtual PICmicro microcontroller, this is a simulation tool that allows users to write and execute MPASM assembler code for the PIC16F84 microcontroller on-screen. Using this you can actually see what happens inside the PICmicro MCU as each instruction is executed, which enhances understanding.

- Comprehensive instruction through 45 tutorial sections
- Includes Vlab, a Virtual PICmicro microcontroller: a fully functioning simulator
- Tests, exercises and projects covering a wide range of PICmicro MCU applications
- Includes MPLAB assembler
- Visual representation of a PICmicro showing architecture and functions
- Expert system for code entry helps first time users
- Shows data flow and fetch execute cycle and has challenges (washing machine, lift, crossroads etc.)
- Imports MPASM files.

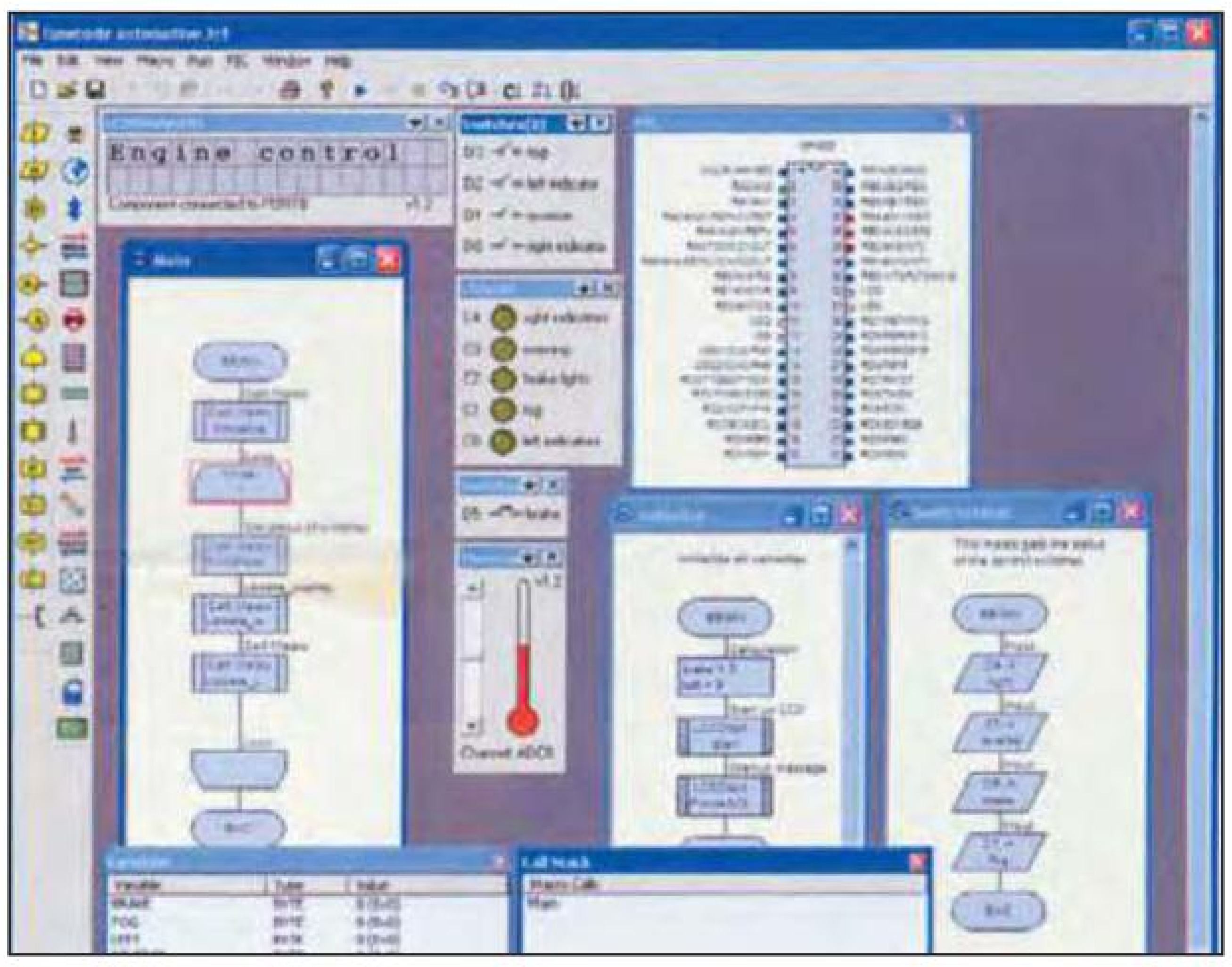


FLOWCODE FOR PICmicro V8

Flowcode is a very high level language programming system based on flowcharts. Flowcode allows you to design and simulate complex systems in a matter of minutes. A powerful language that uses macros to facilitate the control of devices like 7-segment displays, motor controllers and LCDs. The use of macros allows you to control these devices without getting bogged down in understanding the programming. When used in conjunction with the development board this provides a seamless solution that allows you to program chips in minutes.

- Requires no programming experience
- Allows complex PICmicro applications to be designed quickly
- Uses international standard flow chart symbols
- Full on-screen simulation allows debugging and speeds up the development process.
- Facilitates learning via a full suite of demonstration tutorials
- Produces code for a wide range of devices
- 16-bit arithmetic strings and string manipulation
- Pulse width modulation
- I2C.

Please note: Due to popular demand, Flowcode is now available as a download. Please include your email address and a username (of your choice) on your order. A unique download code will then be emailed to you.



PRICES

Prices for each of the CD-ROMs above are:

(Order form on next page)

(UK and EU customers add VAT to ‘plus VAT’ prices)

Single License	**£99**	**plus VAT**
Site Licence	**£499**	**plus VAT**
Flowcode **(choose PIC-8b, PIC-16b, PIC-32b, AVR/Arduino,ARM)**	**Contact us for pricing**	

ELECTRONICS TEACH-IN 2

ELECTRONICS TEACH-IN 2 CD-ROM
USING PIC MICROCONTROLLERS A PRACTICAL INTRODUCTION

This *Teach-In* series of articles was originally published in *EPE* in 2008 and, following demand from readers, has been collected together in the *Electronics Teach-In 2* CD-ROM.

The series is aimed at those using PIC microcontrollers for the first time. Each part of the series includes breadboard layouts to aid understanding and a simple programmer project is provided.

Also included are 29 *PIC N' Mix* articles, also republished from *EPE*. These provide a host of practical programming and interfacing information, mainly for those that have already got to grips with using PIC microcontrollers. An extra four part beginners guide to using the C programing language for PIC microcontrollers is also included.

The CD-ROM also contains all of the software for the *Teach-In 2* series and *PIC N' Mix* articles, plus a range of items from Microchip – the manufacturers of the PIC microcontrollers. The material has been compiled by Wimborne Publishing Ltd. with the assistance of Microchip Technology Inc.

CD-ROM **Order code ETI2 CD-ROM** **£9.50**

ELECTRONICS TEACH-IN 3

ELECTRONICS TEACH-IN 3 CD-ROM

The three sections of this CD-ROM cover a very wide range of subjects that will interest everyone involved in electronics, from hobbyists and students to professionals. The first 80-odd pages of Teach-In 3 are dedicated to *Circuit Surgery*, the regular *EPE* clinic dealing with readers' queries on circuit design problems – from voltage regulation to using SPICE circuit simulation software.

The second section – *Practically Speaking* – covers the practical aspects of electronics construction. Again, a whole range of subjects, from soldering to avoiding problems with static electricity and indentifying components, are covered. Finally, our collection of *Ingenuity Unlimited* circuits provides over 40 circuit designs submitted by the readers of *EPE*.

The CD-ROM also contains the complete *Electronics Teach-In 1* book, which provides a broad-based introduction to electronics in PDF form, plus interactive quizzes to test your knowledge, TINA circuit simulation software (a limited version – plus a specially written TINA Tutorial).

The Teach-In 1 series covers everything from Electric Current through to Microprocessors and Microcontrollers and each part includes demonstration circuits to build on breadboards or to simulate on your PC.

CD-ROM **Order code ETI3 CD-ROM** **£8.50**

ELECTRONICS TEACH-IN 4

ELECTRONICS TEACH-IN 4 CD-ROM
A BROAD-BASED INTRODUCTION TO ELECTRONICS.

The Teach-In 4 CD-ROM covers three of the most important electronics units that are currently studied in many schools and colleges. These include, Edexcel BTEC level 2 awards and the electronics units of the Diploma in Engineering, Level 2.

The CD-ROM also contains the full Modern Electronics Manual, worth £29.95. The Manual contains over 800 pages of electronics theory, projects, data, assembly instructions and web links.

A package of exceptional value that will appeal to all those interested in learning about electronics or brushing up on their theory, be they hobbyists, students or professionals.

CD-ROM **Order code ETI4 CD-ROM** **£8.99**

ELECTRONICS TEACH-IN BUNDLE – FOR PARTS 3, 4 & 5

CD-ROM **Order code ETIB3 CD-ROM** **£18.95**

ORDERING

ALL PRICES INCLUDE UK POSTAGE

Standard/Student/Basic (Hobbyist) Version price includes postage to most countries in the world EU residents outside the UK add £5 for airmail postage per order

ELECTRONICS TEACH-IN 5

ELECTRONICS TEACH-IN 5
JUMP START

FREE CD-ROM

15 design and build circuit projects dedicated to newcomers or those following courses in school and colleges. The projects are: Moisture Detector, Quiz Machine, Battery Voltage Checker, Solar-Powered Charger, Versatile Theft Alarm, Spooky Circuits, Frost Alarm, Mini Christmas Lights, iPod Speaker, Logic Probe, DC Motor Controller, Egg Timer, Signal Injector Probe, Simple Radio Receiver, Temperature Alarm.

PLUS:

PIC' N MIX – starting out with PIC Microcontrollers and PRACTICALLY SPEAKING – the techniques of project construction.

FREE CD-ROM – The free CD-ROM is the complete Teach-In 2 book providing a practical introduction to PIC Microprocessors plus MikroElektronika, Microchip and L-Tek PoScope software.

160 Pages **Order code ETI5** **£8.99**

Please send me: **CD-ROM ORDER FORM**

☐ **Assembly for PICmicro V6**
☐ **'C' for 16 Series PICmicro V6**

Version required:
☐ **Single Licence**
☐ **Site licence**

Note: The software on each version is the same, only the licence for use varies.

☐ **PICmicro Multiprogrammer Board and Development Board (hardware)**

☐ **Circuit Wizard – Standard**
☐ **Circuit Wizard – Professional**

☐ **GCSE Electronics**

☐ **TINA Design Suite V11 Basic (Hobbyist)**
☐ **TINA Design Suite V11 (Student)**

☐ **Teach-In 2**
☐ **Teach-In 3**
☐ **Teach-In 4**
☐ **Teach-In Bundle (3, 4 and 5)**

Full name . . . .

Address . . . .

. . . . Post code . . . . Tel . . . .

Email: . . . .

☐ I enclose cheque/PO in £ sterling payable to Practical Electronics for £ . . . .
☐ Please charge my Visa/Mastercard: £ . . . .

Valid From: . . . . Card expiry date: . . . .

Card No: . . . .

Card Security Code . . . .

(The last 3 digits on or just under the signature strip)

Single License and Site License Versions – overseas readers add £5 to the basic price of each order for airmail postage (**do not add VAT** unless you live in an EU (European Union) country, then add VAT at 20% or provide your official VAT registration number).

Send your order to:
Wimborne Publishing Ltd
113 Lynwood Drive
Wimborne
Dorset BH21 1UU

To order by phone call:

01202 880299. Fax: 01202 843233

Goods are normally sent within seven days

E-mail: stewart.kearn@wimborne.co.uk

Online shop:

www.epemag.com

DIRECT BOOK SERVICE

Teach-In 2017

Introducing the BBC micro:bit

IN FULL COLOUR
For Beginners of Any Age
Python Coding on the BBC micro:bit
Jim Gatenby

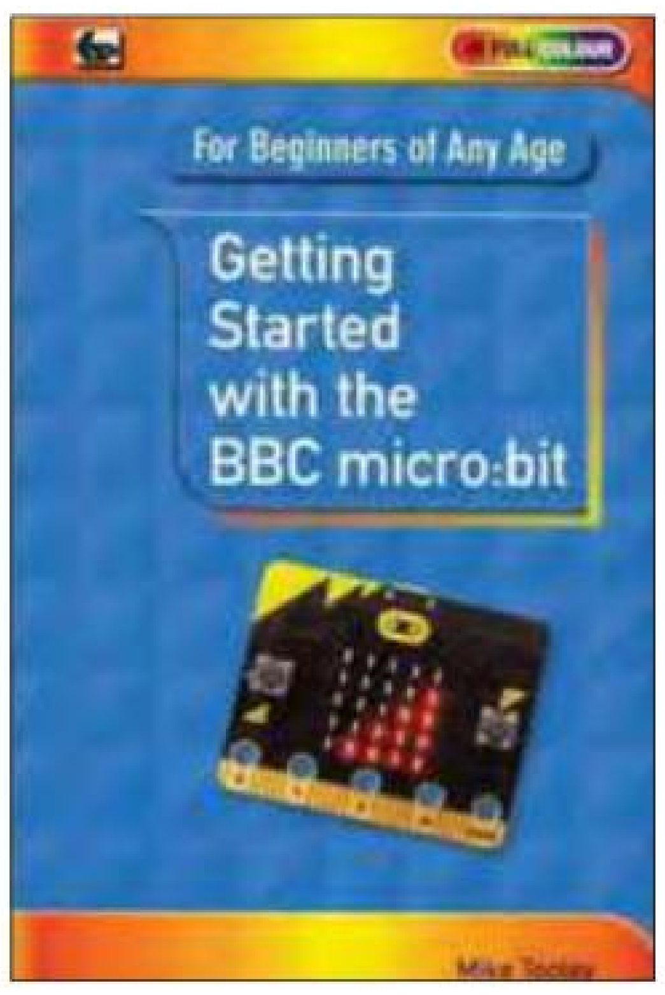
PYTHON CODING ON THE BBC MICRO:BIT
Jim Gatenby

Python is the leading programming language, easy to learn and widely used by professional programmers. This book uses MicroPython, a version of Python adapted for the BBC Micro:bit.

Among the many topics covered are: main features of the BBC micro:bit including a simulation in a web browser screen; various levels of programming languages; Mu Editor for writing, saving and retrieving programs, with sample programs and practice exercises; REPL, an interactive program for quickly testing lines of code; scrolling messages, creating and animating images on the micro:bit's LEDs; playing and creating music, sounds and synthesized speech; using the on-board accelerometer to detect movement of the micro:bit on three axes; glossary of computing terms.

This book is written using plain English, avoids technical jargon wherever possible and covers many of the coding instructions and methods which are common to most programming languages. It should be helpful to beginners of any age, whether planning a career in computing or writing code as an enjoyable hobby.

118 Pages **Order code PYTH MBIT** £7.99



GETTING STARTED WITH THE BBC MICRO:BIT
Mike Tooley

Not just an educational resource for teaching youngsters coding, the BBC micro:bit is a tiny low cost, low-profile ARM-based single-board computer. The board measures 43mm × 52mm but despite its diminutive footprint it has all the features of a fully fledged microcontroller together with a simple LED matrix display, two buttons, an accelerometer and a magnetometer.

Mike Tooley's book will show you how the micro:bit can be used in a wide range of applications from simple domestic gadgets to more complex control systems such as those used for lighting, central heating and security applications. Using Microsoft Code Blocks, the book provides a progressive introduction to coding as well as interfacing with sensors and transducers.

Each chapter concludes with a simple practical project that puts into practice what the reader has learned. The featured projects include an electronic direction finder, frost alarm, reaction tester, battery checker, thermostatic controller and a passive infrared (PIR) security alarm.

No previous coding experience is assumed, making this book ideal for complete beginners as well as those with some previous knowledge. Self-test questions are provided at the end of each chapter, together with answers at the end of the book. So whatever your starting point, this book will take you further along the road to developing and coding your own real-world applications.

108 Pages **Order code BBC MBIT** £7.99

THEORY AND REFERENCE

PRACTICAL ELECTRONICS HANDBOOK – 6th Ed
Ian Sinclair
440 pages **Order code NE21** £33.99

STARTING ELECTRONICS – 4th Ed
Keith Brindley
296 pages **Order code ELSEV100** £18.99

ELECTRONIC CIRCUITS – FUNDAMENTALS & APPLICATIONS – Updated version
Mike Tooley
400 pages **Order code TF43** £32.99

FUNDAMENTAL ELECTRICAL AND ELECTRONIC PRINCIPLES – 3rd Ed
C.R. Robertson
368 pages **Order code TF47** £21.99

A BEGINNER'S GUIDE TO TTL DIGITAL ICs
Robert Penfold
142 pages **OUT OF PRINT BP332** £5.45

UNDERSTANDING ELECTRONIC CONTROL SYSTEMS
Owen Bishop
228 pages **Order code NE35** £36.99

The books listed here have been selected by the *Practical Electronics* editorial staff as being of special interest to everyone involved in electronics and computing. They are supplied by mail order direct to your door.

FOR A FULL DESCRIPTION OF THESE BOOKS AND CD-ROMS SEE THE SHOP ON OUR WEBSITE

www.epemag.com

All prices include UK postage

MICROPROCESSORS

INTERFACING PIC MICROCONTROLLERS – 2nd Ed
Martin Bates
298 pages **Order code NE48** £34.99

PROGRAMMING 16-BIT PIC MICROCONTROLLERS IN C – LEARNING TO FLY THE PIC24
Lucio Di Jasio (Application Segments Manager, Microchip, USA)
496 pages + CD-ROM **Order code NE45** £38.00

INTRODUCTION TO MICROPROCESSORS AND MICROCONTROLLERS – 2nd Ed
John Crisp
222 pages **Order code NE31** £29.99

THE PIC MICROCONTROLLER YOUR PERSONAL INTRODUCTORY COURSE – 3rd Ed
John Morton
270 pages **Order code NE36** £25.00

PIC IN PRACTICE – 2nd Ed
David W. Smith
308 pages **Order code NE39** £24.99

MICROCONTROLLER COOKBOOK
Mike James
240 pages **Order code NE26** £36.99

BOOK ORDERING DETAILS

All prices include UK postage.

For airmail, add £3 per book to Europe, £4 for rest of the world per book.

CD-ROM prices include VAT and/or postage to anywhere in the world.

Send a cheque, (£ sterling only) made payable to: **Practical Electronics** or credit card details (Visa or Mastercard) to:

WIMBORNE PUBLISHING LIMITED,
113 LYNWOOD DRIVE, WIMBORNE, DORSET BH21 1UU

Books are normally sent within seven days of receipt of order.

Please check price (see latest issue of Practical Electronics or website) before ordering from old lists.

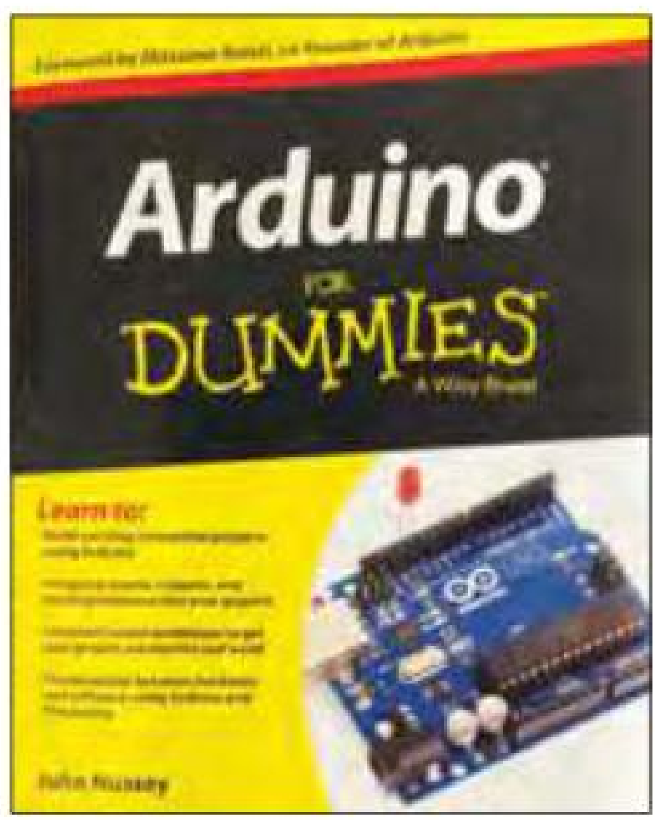
For a full description of these books please see the shop on our website.
Tel: 01202 880299 – E-mail: stewart.kearn@wimborne.co.uk

Order from our online shop at: www.epemag.com

ARDUINO

Teach-In 2016

See opposite for our popular introduction to the Arduino

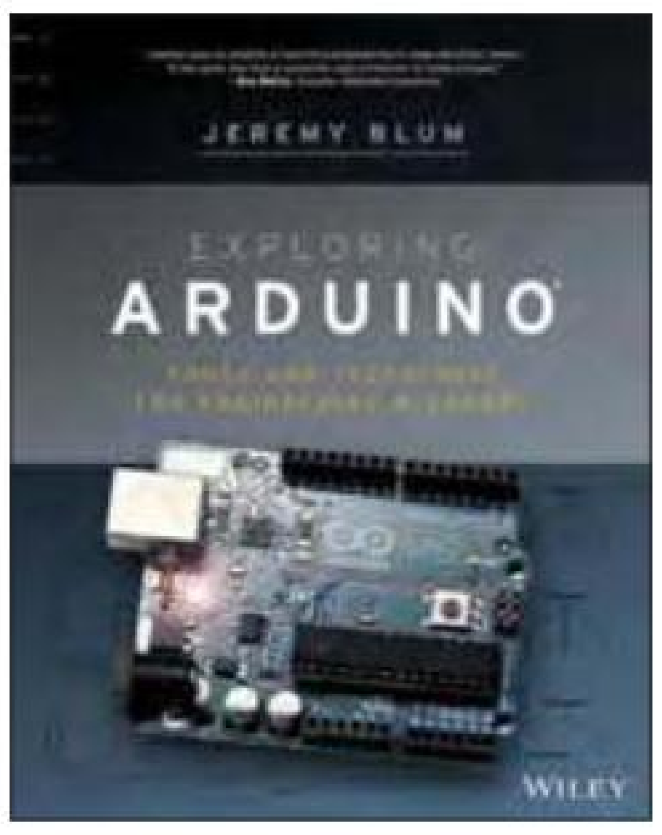


ARDUINO FOR DUMMIES
John Nussey

Arduino is no ordinary circuit board. Whether you're an artist, a designer, a programmer, or a hobbyist, Arduino lets you learn about and play with electronics. You'll discover how to build a variety of circuits that can sense or control real-world objects, prototype your own product, and even create interactive artwork. This handy guide is exactly what you need to build your own Arduino project – what you make is up to you!

- **Learn by doing** – start building circuits and programming your Arduino with a few easy examples – right away!
- **Easy does it** – work through Arduino sketches line by line, and learn how they work and how to write your own.
- **Solder on!** – don't know a soldering iron from a curling iron? No problem! You'll learn the basics and be prototyping in no time.
- **Kitted out** – discover new and interesting hardware to turn your Arduino into anything from a mobile phone to a Geiger counter.
- **Become an Arduino savant** – find out about functions, arrays, libraries, shields and other tools that let you take your Arduino project to the next level
- **Get social** – teach your Arduino to communicate with software running on a computer to link the physical world with the virtual world

438 Pages Order code ARDDUM01 £19.99



EXPLORING ARDUINO
Jeremy Blum

Arduino can take you anywhere. This book is the roadmap.

Exploring Arduino shows how to use the world's most popular microcontroller to create cool, practical, artistic and educational projects. Through lessons in electrical engineering, programming and human-computer interaction, this book walks you through specific, increasingly complex projects, all the while providing best practices that you can apply to your own projects once you've mastered these. You'll acquire valuable skills – and have a whole lot of fun.

- **Explore** the features of commonly used Arduino boards
- **Use** Arduino to control simple tasks or complex electronics
- **Learn** principles of system design, programming and electrical engineering
- **Discover** code snippets, best practices and system schematics you can apply to your original projects
- **Master** skills you can use for engineering endeavours in other fields and with different platforms

357 Pages Order code EXPARD01 £26.99

COMPUTING AND ROBOTICS

NEWNES INTERFACING COMPANION
Tony Fischer-Cripps
295 pages Order code NE38 £41.00

HOW TO BUILD A COMPUTER MADE EASY
Robert Penfold
120 pages Order code BP707 £8.49

EASY PC CASE MODDING
Robert Penfold
192 pages + CD-ROM Order code BP542 £8.99

FREE DOWNLOADS TO PEP-UP AND PROTECT YOUR PC
Robert Penfold
128 pages Order code BP722 £7.99

WINDOWS XP EXPLAINED
N. Kantaris and P.R.M. Oliver
264 pages Order code BP514 £7.99

THE INTERNET – TWEAKS, TIPS AND TRICKS
Robert Penfold
128 pages Order code BP721 £7.99

eBAY – TWEAKS, TIPS AND TRICKS
Robert Penfold
128 pages Order code BP716 £7.50

AN INTRODUCTION TO eBAY FOR THE OLDER GENERATION
Cherry Nixon
120 pages Order code BP709 £8.49

HOW TO FIX YOUR PC PROBLEMS
Robert Penfold
128 pages Order code BP705 £8.49

AN INTRODUCTION TO WINDOWS VISTA
P.R.M. Oliver and N. Kantarris
120 pages Order code BP703 £8.49

WINDOWS 8.1 EXPLAINED
180 Pages Order code BP747 £10.99

AN INTRODUCTION TO THE NEXUS 7
118 Pages Order code BP744 £8.99

COMPUTING FOR THE OLDER GENERATION
Jim Gatenby
308 pages Order code BP601 £8.99

ANDROIDS, ROBOTS AND ANIMATRONS
Second Edition – John Iovine
224 pages Order code MGH1 £16.99

ROBOT BUILDERS COOKBOOK
Owen Bishop
366 pages Order code NE46 £26.00

INTRODUCING ROBOTICS WITH LEGO MINDSTORMS
Robert Penfold
288 pages + Order code BP901 £14.99

MORE ADVANCED ROBOTICS WITH LEGO MINDSTORMS
Robert Penfold
298 pages Order code BP902 £14.99

HOW TO FIX YOUR PC PROBLEMS
Robert Penfold
128 pages Order code BP705 £8.49

WINDOWS 7 – TWEAKS, TIPS AND TRICKS
Andrew Edney
120 pages Order code BP708 £8.49

GETTING STARTED IN COMPUTING FOR THE OLDER GENERATION
Jim Gatenby
120 pages Order code BP704 £8.49

WINDOWS 8.1 EXPLAINED
Noel Kantaris
180 Pages Order code BP747 £10.99

COMPUTING WITH A LAPTOP FOR THE OLDER GENERATION
Robert Penfold
120 pages Order code BP702 £8.49

AN INTRODUCTION TO EXCEL SPREADSHEETS
Jim Gatenby
18 pages Order code BP701 £8.49

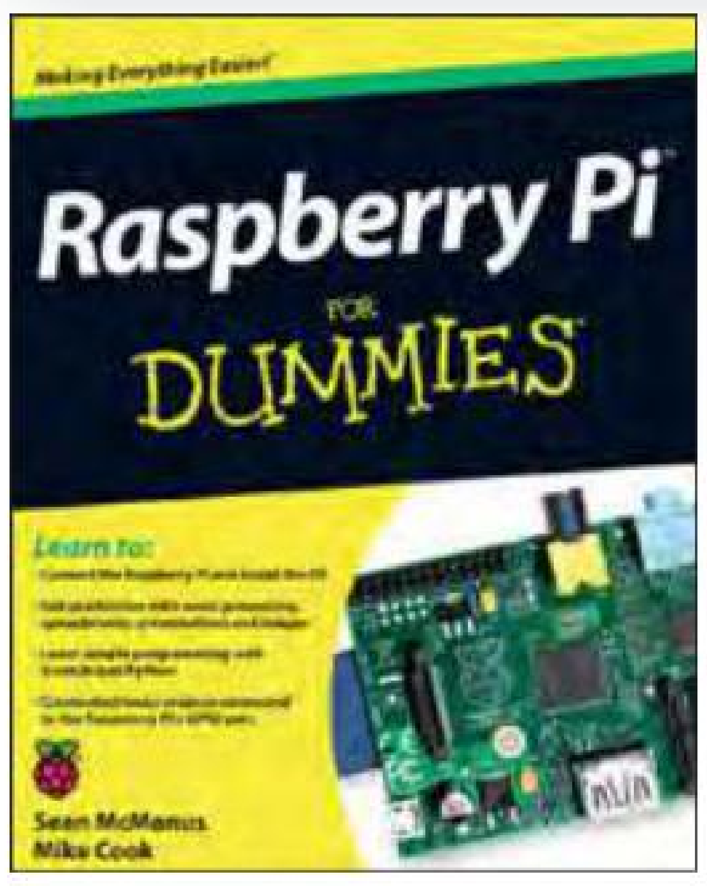
KINDLE FIRE HDX EXPLAINED
118 Pages Order code BP743 £8.99

AUDIO & VIDEO

VALVE AMPLIFIERS – 4th Ed
Morgan Jones
288 pages Order code ELSEV33 £46.99

BUILDING VALVE AMPLIFIERS
Morgan Jones
368 pages Order code NE40 £29.00

RASPBERRY PI



RASPBERRY Pi FOR DUMMIES
Sean McManus and Mike Cook

Write games, compose and play music, even explore electronics – it's easy as Pi! The Raspberry Pi offers a plateful of opportunities, and this great resource guides you step-by-step, from downloading, copying, and installing the software to learning about Linux and finding cool new programs for work, photo editing, and music. You'll discover how to write your own Raspberry Pi programs, create fun games, and much more!

Open this book and find:
What you can do with Python; Ways to use the Raspberry Pi as a productivity tool; How to surf the web and manage files; Secrets of Sonic Pi music programming; A guide to creating animations and arcade games; Fun electronic games you can build; How to build a 3D maze in Minecraft; How to play music and videos on your Raspberry Pi.

400 Pages Order code RPiDUM01 £17.99

RASPBERRY Pi MANUAL: A practical guide to the revolutionary small computer
176 pages Order code H001 £17.99

RASPBERRY Pi USER-GUIDE – 4th Ed
262 pages Order code JW001 £20.90

PROGRAMMING THE RASPBERRY Pi
192 pages Order code MGH4 £10.99

GETTING STARTED WITH RASPBERRY Pi
164 pages Order code OR01 £11.50

TEACH-IN BOOKS

ELECTRONICS TEACH-IN 6

ELECTRONICS TEACH-IN 6
RASPBERRY Pi
A COMPREHENSIVE GUIDE TO RASPBERRY Pi

ONLY AVAILABLE ON CD-ROM

ELECTRONICS TEACH-IN 6
A COMPREHENSIVE GUIDE TO RASPBERRY Pi

Mike & Richard Tooley

Teach-In 6 contains an exciting series of articles that provides a complete introduction to the Raspberry Pi, the low-cost computer that has taken the education and computing world by storm.

This latest book in our Teach-In series will appeal to electronic enthusiasts and computer buffs wanting to get to grips with the Raspberry Pi.

Anyone considering what to do with their Pi, or maybe they have an idea for a project but don't know how to turn it into reality, will find Teach-In 6 invaluable. It covers: Programming, Hardware, Communications, Pi Projects, Pi Class, Python Quickstart, Pi World, and Home Baking.

The CD-ROM also contains all the necessary software for the series so that readers can get started quickly and easily with the projects and ideas covered.

160 Pages **Order code ETI6** **£8.99**

ELECTRONICS TEACH-IN 7 (Includes free CD-ROM)

ELECTRONICS TEACH-IN 7
DISCRETE LINEAR CIRCUIT DESIGN

RUNNING LOW – ORDER NOW!

ELECTRONICS TEACH-IN 7
DISCRETE LINEAR CIRCUIT DESIGN

Mike & Richard Tooley

Teach-In 7 is a complete introduction to the design of analogue electronic circuits. Ideal for everyone interested in electronics as a hobby and for those studying technology at schools and colleges. Supplied with a free cover-mounted CD-ROM containing all the circuit software for the course, plus demo CAD software for use with the Teach-In series

Discrete Linear Circuit Design* Understand linear circuit design* Learn with 'TINA' – modern CAD software* Design simple, but elegant circuits* Five projects to build: Pre-amp, Headphone Amp, Tone Control, VU-meter, High Performance Audio Power Amp.

PLUS

Audio Out – an analogue expert's take on specialist circuits
Practically Speaking – the techniques of project building

160 Pages **Order code ETI7** **£8.99**

ELECTRONICS TEACH-IN 8

ELECTRONICS TEACH-IN 8
INTRODUCING THE ARDUINO

ONLY AVAILABLE ON CD-ROM

ELECTRONICS TEACH-IN 8
INTRODUCING THE ARDUINO

Mike & Richard Tooley

Hardware – learn about components and circuits; Programming – powerful integrated development system; Microcontrollers – understand control operations; Communications – connect to PCs and other Arduinos

This exciting series has been designed for electronics enthusiasts who want to get to grips with the inexpensive, immensely popular Arduino microcontroller, as well as coding enthusiasts who want to explore hardware and interfacing. Teach-In 8 will provide a one-stop source of ideas and practical information.

The Arduino offers a remarkably effective platform for developing a huge variety of projects; from operating a set of Christmas tree lights to remotely controlling a robotic vehicle through wireless or the Internet. Teach-In 8 is based around a series of practical projects with plenty of information to customise each project.

This book also includes *PIC n' Mix*: PICs and the PICkit 3 - A Beginners guide by Mike O'Keefe and *Circuit Surgery* by Ian Bell - State Machines part 1 and 2.

The CD-ROM includes files for Teach-In 8 plus Microchip MPLAB IDE XC8 8-bit Compiler and PICkit 3 User Guide. Also included is Lab-Nation Smartscope software.

160 Pages **Order code ETI8** **£8.99**

VISIT OUR WEBSITE FOR MORE BOOKS AND FAST, EASY ONLINE ORDERING
WWW.EPEMAG.COM

BOOK ORDER FORM

Full name: ....

Address: ....

....

.... Post code: .... Telephone No: ....

Email: ....

☐ I enclose cheque/PO payable to Practical Electronics for £ ....

☐ Please charge my card £ .... Card expiry date...

Card Number ....

Valid From Date .... Card Security Code .... (The last three digits on or just below the signature strip)

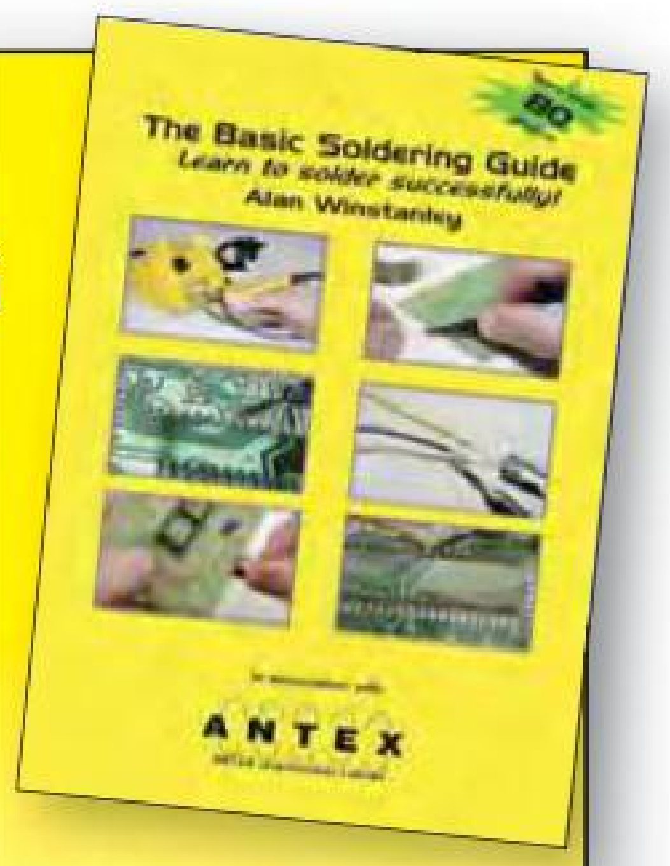
Please send book order codes: ....

....

....

Please continue on separate sheet of paper if necessary

THE BASIC SOLDERING GUIDE



LEARN TO SOLDER SUCCESSFULLY!
ALAN WINSTANLEY

The No.1 resource for learning all the basic aspects of electronics soldering by hand.

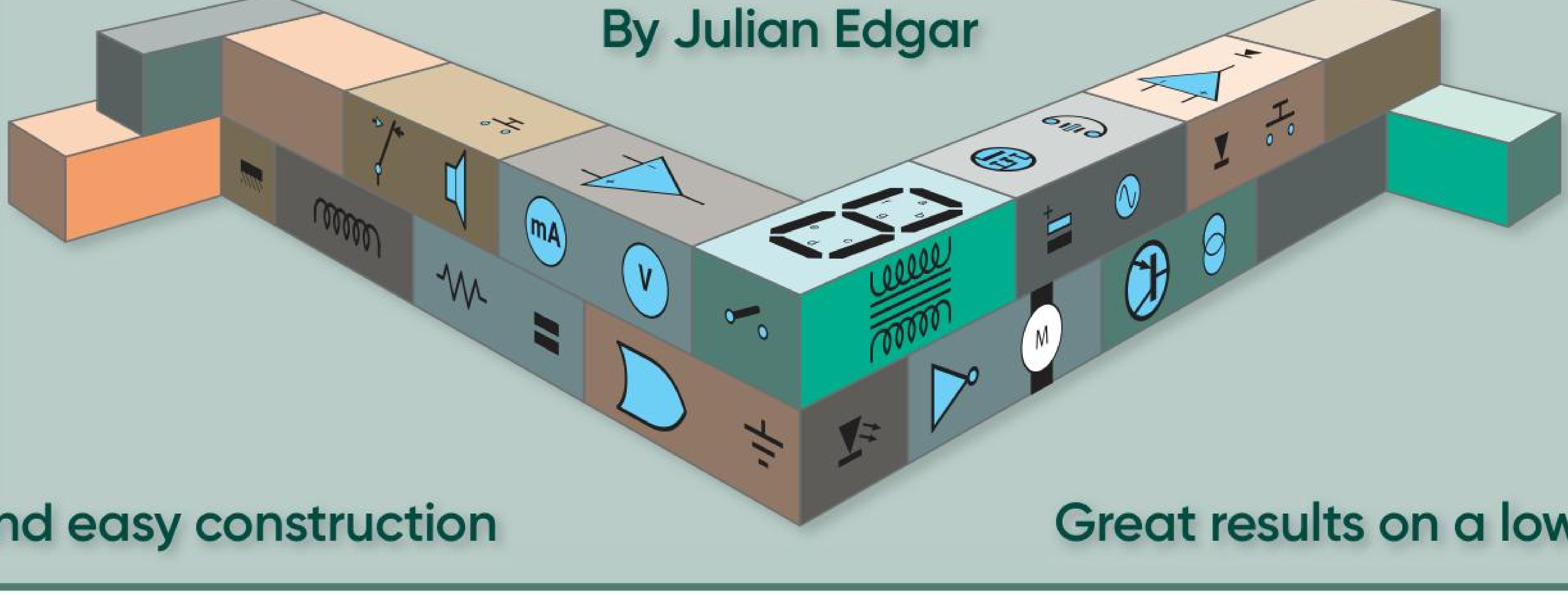
With more than 80 high quality colour photographs, this book explains the correct choice of soldering irons, solder, fluxes and tools. The techniques of how to solder and desolder electronic components are then explained in a clear, friendly and non-technical fashion so you'll be soldering successfully in next to no time! The book also includes sections on reflow soldering and desoldering techniques, potential hazards, useful resources and a very useful troubleshooting guide.

Also ideal for those approaching electronics from other industries, the *Basic Soldering Guide Handbook* is the best resource of its type, and thanks to its excellent colour photography and crystal clear text, the art of soldering can now be learned by everyone!

86 Pages **Order code AW1** **£9.99**

Electronic Building Blocks

By Julian Edgar



Quick and easy construction

Great results on a low budget

Four simple and cheap LED kits

This month, I'm going to cover four modules. They have two things in common – they all use LEDs, and they're all small and simple kits that you build yourself. Oh yes, and another thing – they're all very cheap! They are excellent as a beginner's kits – mini projects that someone new to electronics (either young or old) can assemble, thus learning component recognition and soldering skills.

Sound-triggered flasher

The first kit costs a grand total of just over £1, delivered! It comprises a small printed circuit board (PCB), just 30 × 24mm. A mere handful of components is used – three resistors, two transistors, two electrolytic capacitors, an electret microphone and five bright blue LEDs.

Assembly is straightforward, but if you are new to building kits, remember that the electrolytic capacitors, LEDs, transistors and microphone are all polarised – they must be soldered into place the 'right way around'. The PCB generally shows the correct orientations, but ensure you get the LEDs correctly orientated – see Fig.1.

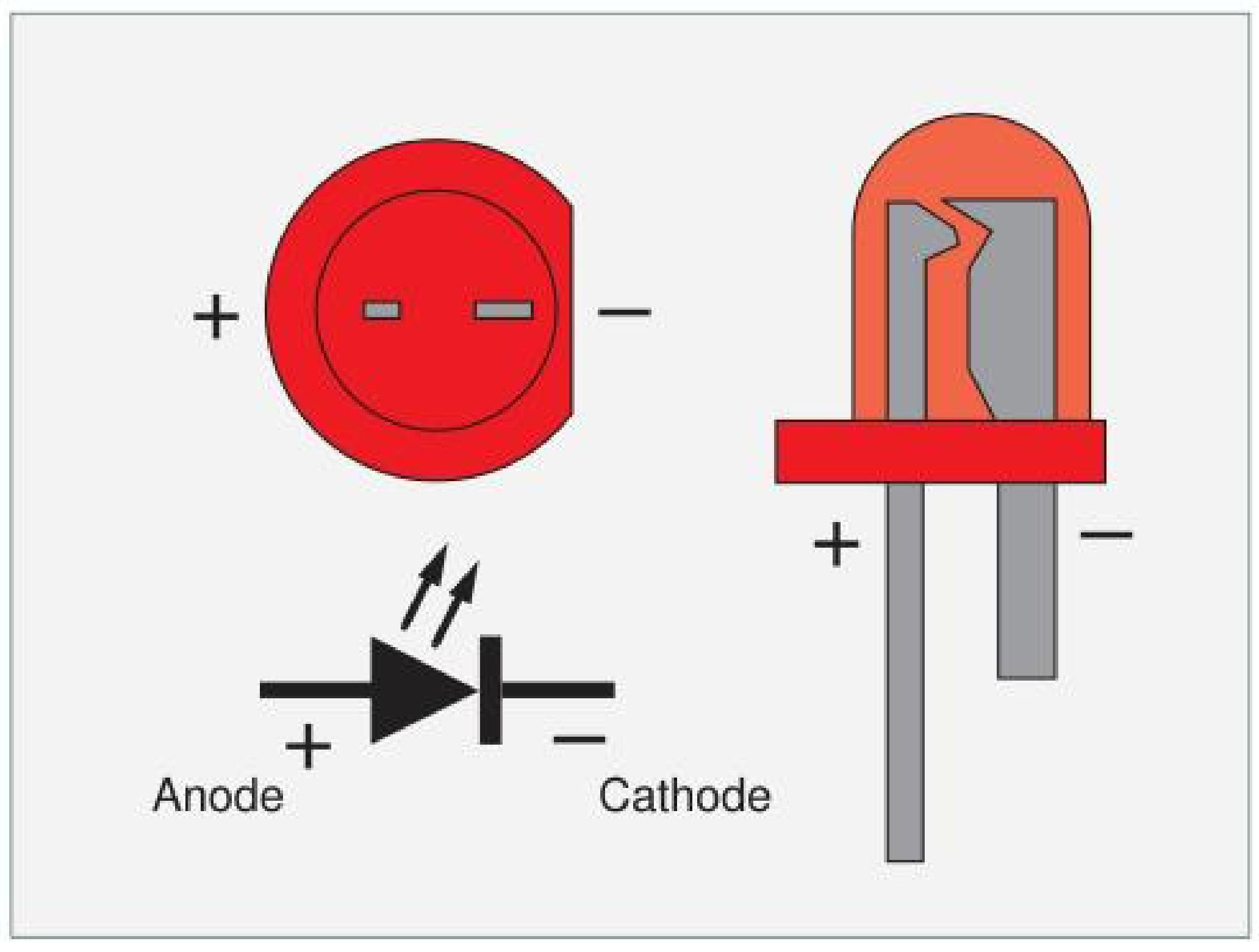


Fig.1. One of the few areas that can be tricky when building these kits is getting the orientation of the LEDs correct. This diagram should help.

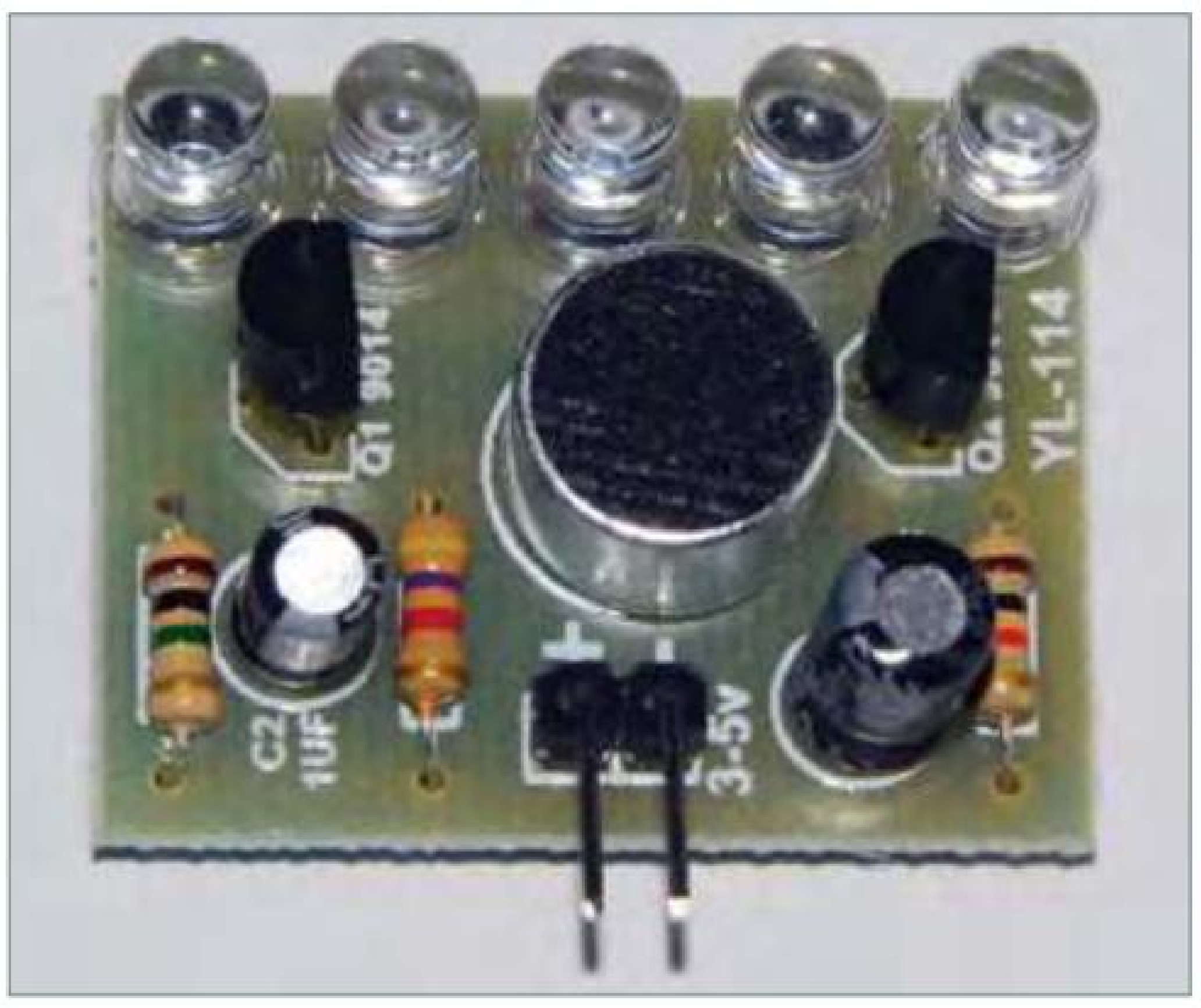


Fig.2. This sound-triggered LED module lights the LEDs when sound is sensed by the on-board microphone. Playing music causes the LEDs to beat in time.

The PCB is nicely laid out with plenty of space for the tip of a soldering iron – and of course, as with all these modules, through-hole (not tiny surface mount) components are used – see Fig.2. The power supply can be anywhere from 3-5V.

So what does the board do? It turns on the LEDs as a group, with the louder the sound sensed by the microphone, the brighter the LEDs. Clap your hands and they light up; speak loudly and they do likewise. Play music, and the LED output modulates with the loudness of the sound – so in this situation, their brightness is always varying.

The module is an ideal addition to a small 5V amplifier (eg, one powered from the USB port).

You'll find the kit by searching on eBay under 'Sound Control 5-LED Melody Rhythm Lamp Module Electronic Production 3V-5.5V'; for example, at the time of writing, item 221811339335.

'Breathing' LED module

This kit is just a little more complex to build than the one above. It has more components and has an integrated circuit (IC). Eight LEDs (again, bright blue) are used, and an adjustment potentiometer (pot) is supplied.

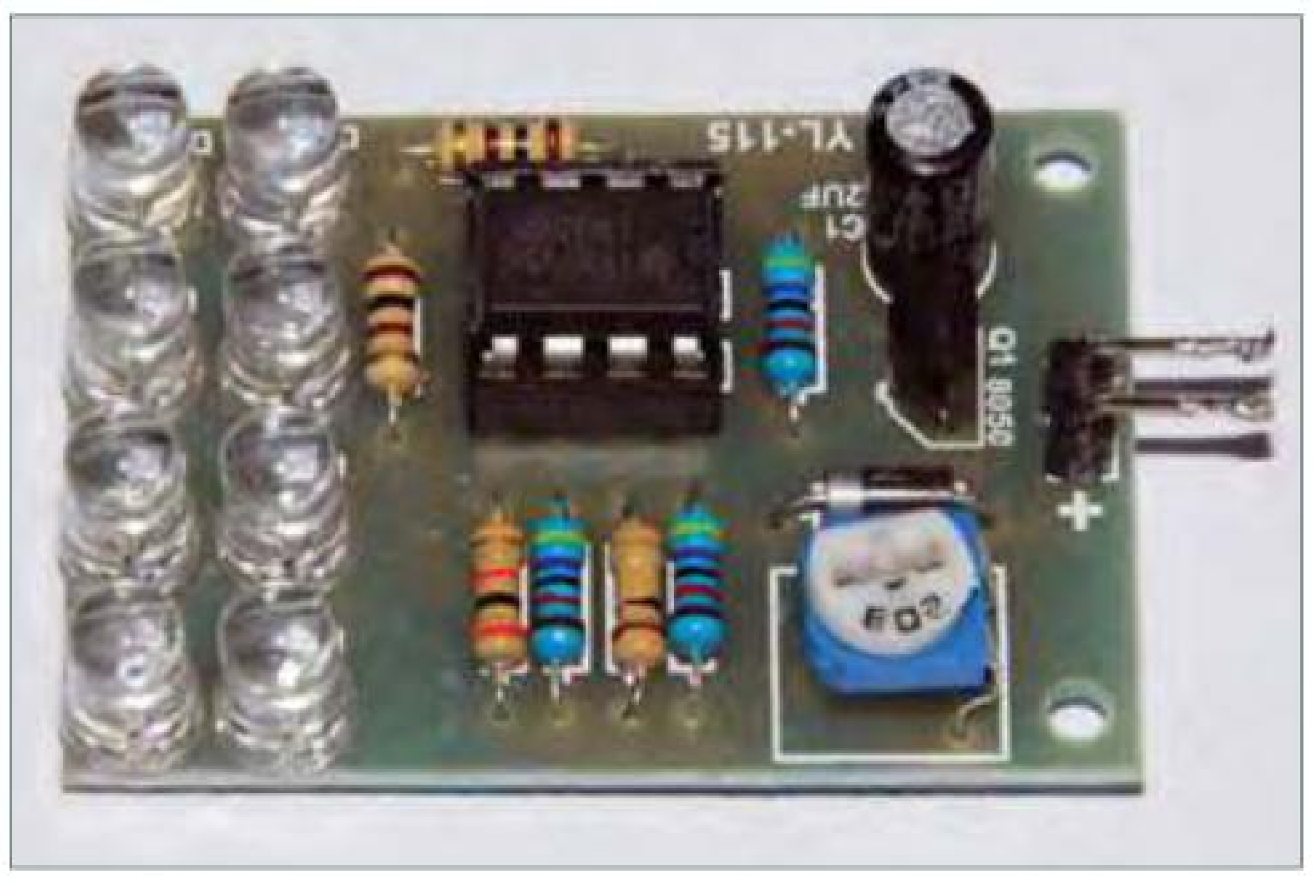


Fig.3. An LED flasher – but not like one you've seen before. The LEDs progressively light and then fade, with the frequency adjusted by the on-board pot. The current draw is quite low, and the LEDs are very bright.

This PCB is also small (42 × 30mm) and it's another that's easy to solder. For beginners, note that a socket is supplied for the IC – this makes it much less likely the IC will be destroyed by an overly generous application of soldering heat!

The price is a neat £1 (delivered). To find it, search eBay under '12V Breathe Light 8-LED Flashing Lamp Parts Electronic DIY Module LM358 Chip'; for example, at the time of writing, item 272160372298.

Power for the board is nominally 12V – however the board functions, at reduced LED brightness, down to about 9V, so it can be run from a (PP3) 9V battery.

So what happens when you connect power? In operation, the LEDs – working as a block – fade from off to full brightness, then back to off again. With the pot at one extreme, full brightness occurs about once per second. With the pot at the other end of its adjustment,

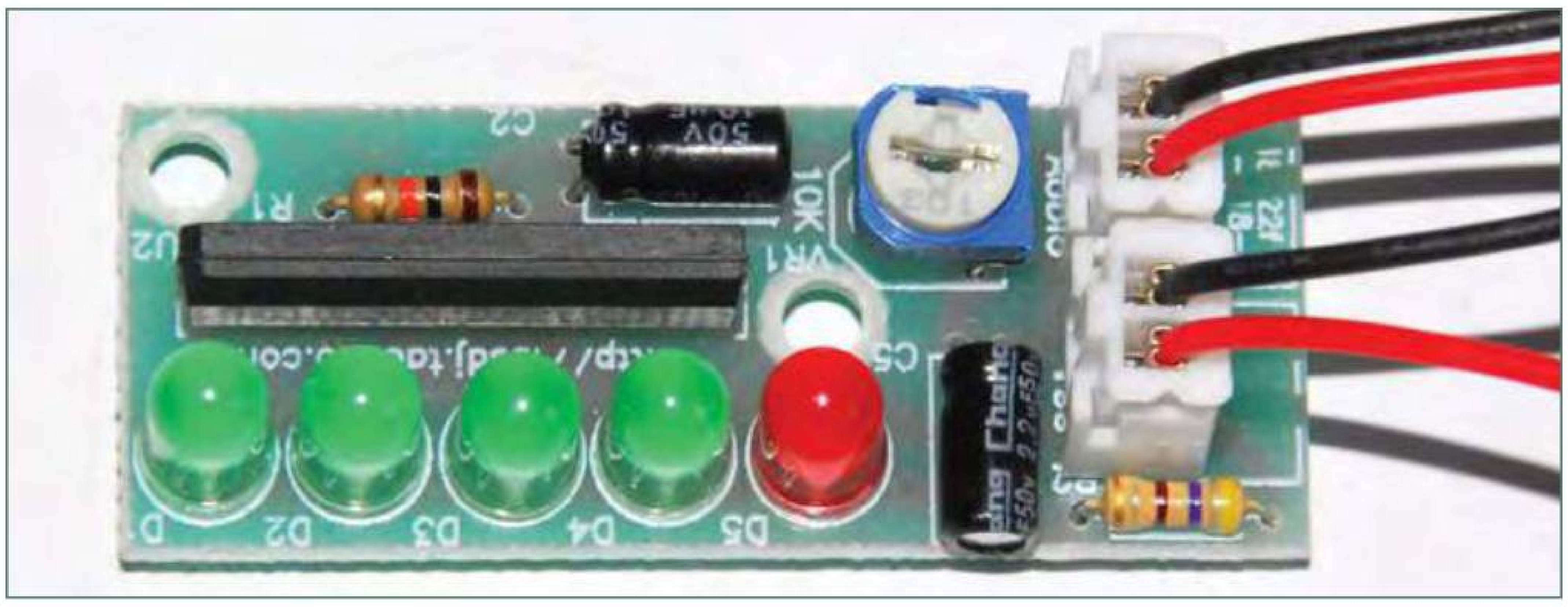


Fig.4. This audio level meter uses a logarithmic display. The adjustment pot allows the module to be used with widely different input levels.

this rate doubles. It's a peculiarly fascinating rhythmic display.

The LEDs are very bright, so all things considered, the board is quite efficient, drawing an average current of just 7mA at 12V.

Uses? The board could be used as an unusual 'equipment on' indicator, to light a display at night (eg, use 10 of them distributed around a shop window), or to internally light equipment like an amplifier equipped with a clear lid. Finally, with such a low current draw, the module could be used as a car or house alarm indicator.

Audio level indicator

Continuing with our progression, where the kits get slightly more difficult each time, I now present the LED Level Meter.

Another tiny PCB (47 × 19mm), this kit uses only a small number of components. So why is it harder? The kit uses an in-line IC that does not come with a socket – so you need to solder quickly and neatly when installing it. Also, ensure you insert it the right way around – the IC is marked with a chamfer on one end, with this marking shown on the PCB overlay. On the kit I built, the 470Ω resistor was not labelled on the PCB – however, the other resistor was labelled, so it's not too hard to work out which is which. Other than that, the rest of the kit remains pretty easy to build, so don't be put off by the description of greater difficulty – it's all relative.

The LED level meter is available from Banggood (ID: 1204427) and costs just under £1.50, delivered.

Two on-board sockets are used, and these come complete with plugs and leads. The first connects to power – anything from 3.5 – 12V. The second connects to an audio source. The adjustment pot allows the audio level that will trigger the red (peak) LED to be varied over a wide range. This means you can use line-level signals (the signal output of a CD player or similar), headphone level signals (like you'd get on the audio jack of a phone) or speaker level signals (the signal feeding the speakers). Note that in all cases, the input to the module is wired in parallel to the audio source – so the music still plays as the meter performs. Interestingly, the KA2284 IC is a logarithmic device, so it works quite well with audio signals that vary over a wide range.

The only downside of the kit is that the LEDs are not very bright – certainly, nothing like as bright as the two modules described above.

Thanks to its wide range of power supply voltage, this is an ideal module to add to any amplifier (remember to use two for stereo).

'Joule thief' torch

The final module that I am going to cover is often referred to as a 'joule thief' torch. It has the amazing ability to work on nearly flat batteries – the sort that people discard.

The module is 50 × 19mm and comes supplied with a small box. However, as you'll see in a moment, I didn't use the box – or some other components. The kit is available from Banggood (ID: 1282949) and costs £1.50, delivered.

The circuit is simple to put together, with the PCB well-labelled. The single AA cell, that the kit is designed to use, is meant to mount on the back of the PCB, and two spring-type battery connectors are supplied to solder to the board. However, in use, I found it much easier to use thin, insulated flying leads and solder these to the pads designed to hold the battery connectors. Run the leads to the battery (really, a single cell) and then you don't have the issue of the cell falling off the board all the time. I didn't much like the box, although I imagine if you try hard enough, you could fit everything in. A small, latching press-button switch is located at one end of the PCB, and the white LED at the other.

There isn't the space here to fully describe how the torch works, but in effect, the LED is run off a high frequency, pulsed supply generated by the oscillating action of the circuit. This allows the LED to light brightly, despite the fact that normally a supply voltage of about 3.5V would be needed to power the white LED. Incredibly, the LED stays bright down to an input voltage of just 0.5V. People usually throw away batteries when the voltage of a 1.5V cell drops to about 1.3V – and at that voltage, this torch works very well.

However, you can't quite get something for nothing, and the current draw is over 130mA at 1.3V – a lot for a small LED torch, so that 'flat' cell won't last for long in continuous operation. Still, it's all pretty intriguing – and at that price, worth buying one just to play with, whether you're a beginner in electronics or not.

Next month

In the next issue, I'll cover a compact and low-cost temperature data logger.

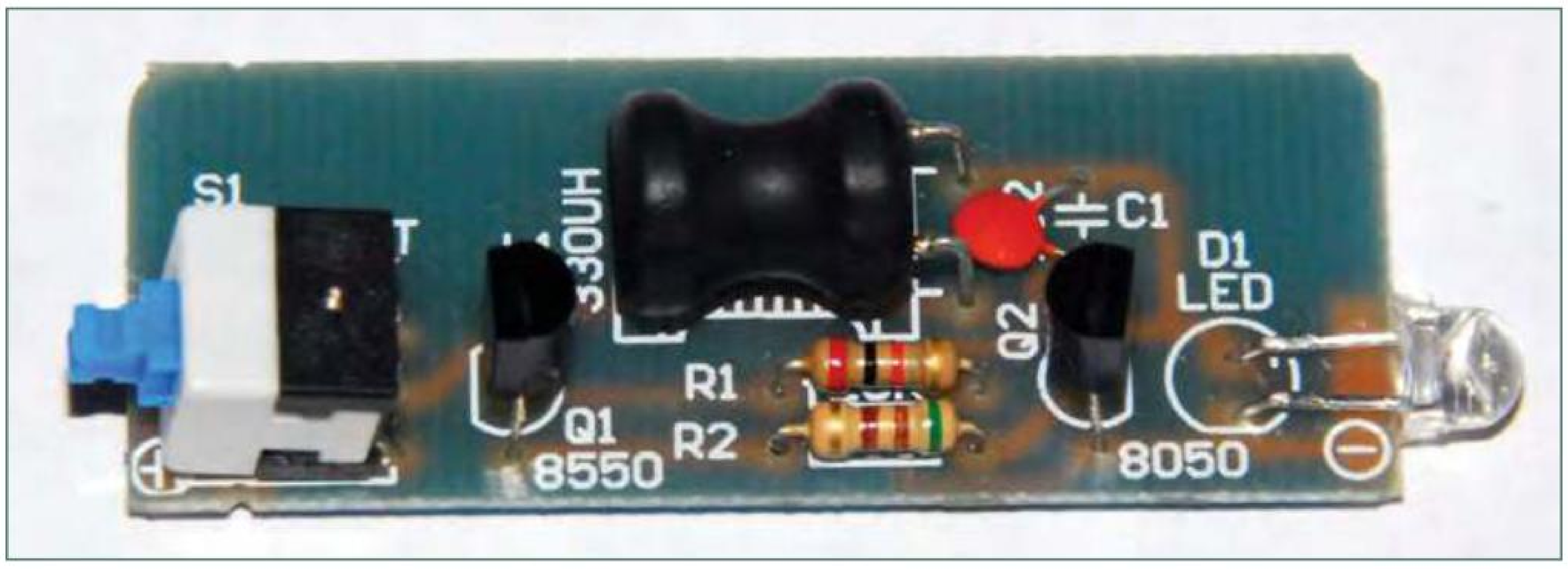


Fig.5. This tiny white LED torch module uses tricky circuitry to operate down to an input voltage of just 0.5V! It will happily run on batteries that you would normally throw away because they are 'flat'.

Data Logger
RC-5
16.3
Elitech

Practical Electronics PCB SERVICE

PROJECT	CODE	PRICE
FEBRUARY 2018		
GPS-Syncronised Analogue Clock Driver	04202171	£12.88
High-Power DC Motor Speed Controller – Part 2		
+ Control Board	11112161	£12.88
+ Power Board	11112162	£15.30
MARCH 2018		
Stationmaster Main Board	09103171	£17.75
+ Controller Board	09103172	
Build the SC200 Amplifier Module – Power Supply	01109111	£16.45
APRIL 2018		
Spring Reverberation Unit	01104171	£15.30
DDS Sig Gen Lid	Black	£8.05
DDS Sig Gen Lid	Blue	£7.05
DDS Sig Gen Lid	Clear	£8.05
MAY 2018		
High Performance RF Prescaler	04112162	£10.45
Micromite BackPack V2	07104171	£10.45
Microbridge	24104171	£5.60
JUNE 2018		
High Performance 10-Octave Stereo Graphic Equaliser	01105171	£15.30
JULY 2018		
Touchscreen Appliance Energy Meter – Part 1	04116061	£17.75
Automotive Sensor Modifier	05111161	£12.88
AUGUST 2018		
Universal Temperature Alarm	03105161	£7.05
Power Supply For Battery-Operated Valve Radios	18108171 18108172 18108173 18108174	£27.50
SEPTEMBER 2018		
3-Way Active Crossover	01108171	£22.60
Ultra-low-voltage Mini LED Flasher	16110161	£5.60
OCTOBER 2018		
6GHz+ Touchscreen Frequency Counter	04110171	£12.88
Two 230VAC MainsTimers	10108161 10108162	£12.88
NOVEMBER 2018		
Super-7 AM Radio Receiver	06111171	£27.50
FEBRUARY 2019		
1.5kW Induction Motor Speed Controller	10105122	£35.00
MARCH 2019		
10-LED Bargraph Main Board	04101181	£11.25
+Processing Board	04101182	£8.60
APRIL 2019		
Heater Controller	10104181	£14.00
MAY 2019		
2× 12V Battery Balancer	14106181	£5.60
Deluxe Frequency Switch	05104181	£10.45
USB Port Protector	07105181	£5.60
JUNE 2019		
Arduino-based LC Meter	04106181	£8.00
USB Flexitimer	19106181	£10.45
JULY 2019		
Full-wave 10A Universal Motor Speed Controller	10102181	£12.90
Recurring Event Reminder	19107181	£8.00
Temperature Switch Mk2	05105181	£10.45
AUGUST 2019		
Brainwave Monitor	25108181	£12.90
Super Digital Sound Effects Module	01107181	£5.60
Watchdog Alarm	03107181	£8.00
PE Theremin (three boards: pitch, volume, VCA)	PETX0819	£19.50
PE Theremin component pack (see p.56, August 2019)	PETY0819	£15.00
OCTOBER 2019		
Programmable GPS-synced Frequency Reference	04107181	£11.50
Digital Command Control Programmer for Decoders	09107181	£8.75
Opto-isolated Mains Relay (main board)	10107181	£11.50
Opto-isolated Mains Relay (2 × terminal extension board)	10107182	
NOVEMBER 2019		
Tinnitus & Insomnia Killer (Jaycar case – see text)	01110181	£8.75
Tinnitus & Insomnia Killer (Altronics case – see text)	01110182	£8.75

PE/EPE PCB SERVICE

Order Code	Project	Quantity	Price
. . . .	. . . .	. . . .	. . . .
. . . .	. . . .	. . . .	. . . .
. . . .	. . . .	. . . .	. . . .
. . . .	. . . .	. . . .	. . . .
. . . .	. . . .	. . . .	. . . .

Name . . . .

Address . . . .

. . . .

Tel . . . .

Email . . . .

I enclose payment of £ . . . . (cheque/PO in **£ sterling only**)

payable to: **Practical Electronics**

Card No . . . .

Valid From . . . . Expiry Date . . . .

Card Security No . . . .

You can also order PCBs by phone, Fax, Email or via the shop on our website on a secure server: www.epemag.com

PCBs for most recent *PE*/*EPE* constructional projects are available. From the July 2013 issue onwards, PCBs with eight-digit codes have silk screen overlays and, where applicable, are double-sided, plated-through hole, with solder mask. They are similar to photos in the project articles. Earlier PCBs are likely to be more basic and may not include silk screen overlay, be single-sided, lack plated-through holes and solder mask.

Always check price and availability in the latest issue or online. A large number of older boards are listed for ordering on our website.

We do not supply kits or components for our projects. For older projects it is important to check the availability of all components before purchasing PCBs.

Back issues of articles are available – see Back Issues page for details.

All prices include VAT and UK p&p. Add £2 per board for airmail outside of Europe. Orders and payment should be sent to:

Practical Electronics, Wimborne Publishing Ltd
113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU
Tel 01202 880299 Fax 01202 843233
Email: stewart.kearn@epemag.wimborne.co.uk
On-line Shop: www.epemag.com

Cheques should be made payable to Practical Electronics **(Payment in £ sterling only)**.

NOTE: While 95% of our boards are held in stock and are dispatched within seven days of receipt of order, please allow a maximum of 28 days for delivery if we need to restock.

CLASSIFIED ADVERTISING

Practical Electronics

If you want your advertisements to be seen by the largest readership at the most economical price our classified page offers excellent value. The rate for semi-display space is £10 (+VAT) per centimetre high, with a minimum height of 2·5cm. All semi-display adverts have a width of 5.5cm. The prepaid rate for classified adverts is 40p (+VAT) per word (minimum 12 words).

Cheques are made payable to Practical Electronics. **VAT must be added**. Advertisements with remittance should be sent to: Practical Electronics, 113 Lynwood Drive, Wimborne, Dorset, BH21 1UU.

Tel 01202 880299 Email: stewart.kearn@wimborne.co.uk

For rates and further information on display and classified advertising please contact our Advertisement Manager, Stewart Kearn – see below.

Practical Electronics reaches more UK readers than any other UK monthly hobby electronics magazine. Our sales figures prove it.

We have been the leading monthly magazine in this market for the last twenty-seven years.

BOWOOD ELECTRONICS LTD
Suppliers of Electronic Components
www.bowood-electronics.co.uk
Unit 10, Boythorpe Business Park, Dock Walk, Chesterfield,
Derbyshire S40 2QR. Sales: 01246 200 222
Send large letter stamp for Catalogue

The British Amateur Electronic Club at:
baec.tripod.com
Has many interesting articles on computers; digital electronics and analogue electronics.

COAST ELECTRONICS
BREAKOUTS-COMPONENTS-
CONTRACT DESIGN-3D PRINTER PARTS-
MUSICAL-MICROCONTROLLERS
WWW.COASTELECTRONICS.CO.UK

MISCELLANEOUS

VALVES AND ALLIED COMPONENTS IN STOCK. Phone for free list. Valves, books and magazines wanted. Geoff Davies (Radio), tel. 01788 574774.

PIC DEVELOPMENT KITS, DTMF kits and modules, CTCSS Encoder and Decoder/Display kits.
Visit www.cstech.co.uk

Electrical Industries Charity (EIC)
We help people working in the electrical, electronics and energy community as well as their family members and retirees.
We use workplace programmes that give the industry access to financial grants and a comprehensive range of free and confidential services.
www.electricalcharity.org

Andrew Kenny – Qualified Patent Agent
EPO UKIPO USPTO
Circuits Electric Machinery Mechatronics
Web: www.akennypatentm.com
Email: Enquiries@akennypatentm.com
Tel: 0789 606 9725

ADVERTISING INDEX

CRICKLEWOOD ELECTRONICS 35
ESR ELECTRONIC COMPONENTS 35
HAMMOND ELECTRONICS Ltd 8
JPG ELECTRONICS 80
MICROCHIP . . . . Cover (ii), Cover (iii), 5
PCBWay 3, 9
PEAK ELECTRONIC DESIGN . . . . Cover (iv)
POLABS D.O.O. 58
QUASAR ELECTRONICS 2
SOUNDTRONICS 3
STEWART OF READING 35
TAG-CONNECT 27
TECHNOBOTS 45

Advertisement offices
Wimborne Publishing Ltd
113 Lynwood Drive
Merley
Wimborne,
Dorset BH21 1UU

Tel 01202 880299
Fax 01202 843233
Email stewart.kearn@wimborne.co.uk
Web www.epemag.com

For editorial contact details see page 7.

Next Month – in the December issue

Extremely Sensitive Magnetometer
It doesn't look like a traditional metal detector. (It's not!) But for ferrous metals, its sensitivity is on a par with – or better than – some of the best commercial designs.

Useless Box
Our projects don't all need to be serious, or even practical. Some of them are whimsical; others – like this one – can be downright useless! Nevertheless, it's good fun.

Stamp-sized digital audio player
The DFPlayer Mini is a low-cost digital audio player module. Despite its size and price, it can do a lot.

DON'T OPERATE THE SWITCH
Useless Box

Four-channel High-current DC Fan and Pump Controller
We originally designed this multi-channel pump and fan speed controller for automotive tasks – but it's so much more than that. Use it anywhere you need to adjust the speed of low-voltage DC fans or other PWM-controlled devices.

Colour Maximite Computer – Part 2
Next month, we will provide full construction details of the Colour Maximite Computer. The end result will be a fully functioning computer that will open up a world of possibilities.

PLUS!
All your favourite regular columns from *Audio Out*, *Cool Beans and Circuit Surgery*, *to Electronic Building Blocks*, *Techno Talk* and *Net Work*.

On sale 7 November 2019

Content may be subject to change

WELCOME TO JPG ELECTRONICS
Selling Electronics in Chesterfield for 29 Years

Open Monday to Friday 9am to 5:30pm
And Saturday 9:30am to 5pm

- Aerials, Satellite Dishes & LCD Brackets
- Audio Adaptors, Connectors & Leads
- BT, Broadband, Network & USB Leads
- Computer Memory, Hard Drives & Parts
- DJ Equipment, Lighting & Supplies
- Extensive Electronic Components
  - *ICs, Project Boxes, Relays & Resistors*
- Raspberry Pi & Arduino Products
- Replacement Laptop Power Supplies
- Batteries, Fuses, Glue, Tools & Lots more...

T: 01246 211 202
E: sales@jpgelectronics.com
JPG Electronics, Shaw's Row,
Old Road, Chesterfield, S40 2RB
W: www.jpgelectronics.com

JPG Electronics

Retail & Trade Welcome • Free Parking • Google St View Tour: S40 2RB

Calling all subscribers!

Practical Electronics

If you are one of our valued subscribers then please note that we have changed the way we send subscription renewal reminders.

Instead of sending you a renewal card, we now print a box on the address sheet that comes with your copy of Practical Electronics.

This box will advise you of the last issue of your current subscription.

To renew, you have three easy choices:

1. Call us on: 01202 880299
2. Visit our website at: www.epemag.com
3. Send a cheque with your details to:

Electron Publishing Limited
113 Lynwood Drive
Wimborne
Dorset BH21 1UU

Published on approximately the first Thursday of each month by Electron Publishing Limited, 1 Buckingham Road, Brighton, East Sussex BN1 3RA. Printed in England by Acorn Web Offset Ltd., Normanton WF6 1TW. Distributed by Seymour, 86 Newman St., London W1T 3EX. Subscriptions UK: £26.99 (6 months); £49.85 (12 months); £94.99 (2 years). EUROPE: airmail service, £30.99 (6 months); £57.99 (12 months); £109.99 (2 years). REST OF THE WORLD: airmail service, £37.99 (6 months); £70.99 (12 months); £135.99 (2 years). Payments payable to 'Practical Electronics', Electron Publishing Limited, 113 Lynwood Drive, Wimborne, Dorset BH21 1UU. Email: stewart.kearn@epemag.wimborne.co.uk. PRACTICAL ELECTRONICS is sold subject to the following conditions, namely that it shall not, without the written consent of the Publishers first having been given, be lent, resold, hired out or otherwise disposed of by way of Trade at more than the recommended selling price shown on the cover, and that it shall not be lent, resold, hired out or otherwise disposed of in a mutilated condition or in any unauthorised cover by way of Trade or affixed to or as part of any publication or advertising, literary or pictorial matter whatsoever.

Development Tool of the Month!

MPLAB® ICD 4 In-circuit Debugger

Part Number
DV164045

Overview:

The new MPLAB® ICD 4 introduces a faster processor and increased RAM to deliver up to twice the speed of ICD 3 for the in-circuit debugging of PIC® MCUs (microcontrollers) and dsPIC® digital signal controllers.

ICD 4 also introduces a wider target voltage range and an optional 1 A of power via an external power supply. For maximum flexibility, MPLAB® ICD 4 features a selectable pull-up/pull-down option to the target interface and programmable adjustment of debugging speed for greater productivity.

Key Features:

- Supports many PIC® MCUs and dsPIC® DSCs
- x2 faster than MPLAB® ICD 3
- Reduced wait time improves debugging productivity
- Simplifies migration between PIC® MCUs
- High-performance 32-bit MCU core
- Increased RAM provides 2 MB of buffer memory
- Wider target supply voltage: 1.20 to 5.5 V
- Optional external power supply for 1 A of power
- Programmable adjustment of debugging speed

Order Your MPLAB® ICD 4 In-circuit Debugger Today!

The Microchip name and logo, PIC and MPLAB are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.
All other trademarks mentioned herein are the property of their respective companies.
© 2019 Microchip Technology Inc. All rights reserved. MEC2298-ENG-07-19

PEAK®
electronic design ltd

Celebrating the return of Practical Electronics!
Limited time offer only

Use code PE10 during checkout
(or on the phone!)

DCA75

Advanced Semiconductor Analyser and Curve Tracer

atlas DCA Pro
DCA75
USB + Graphics
VBE=1.216V
at IB=3.6mA
ICLeak=0.000mA
PEAK
electronic design ltd
on - test
scroll - off
warning: avoid charge/voltages

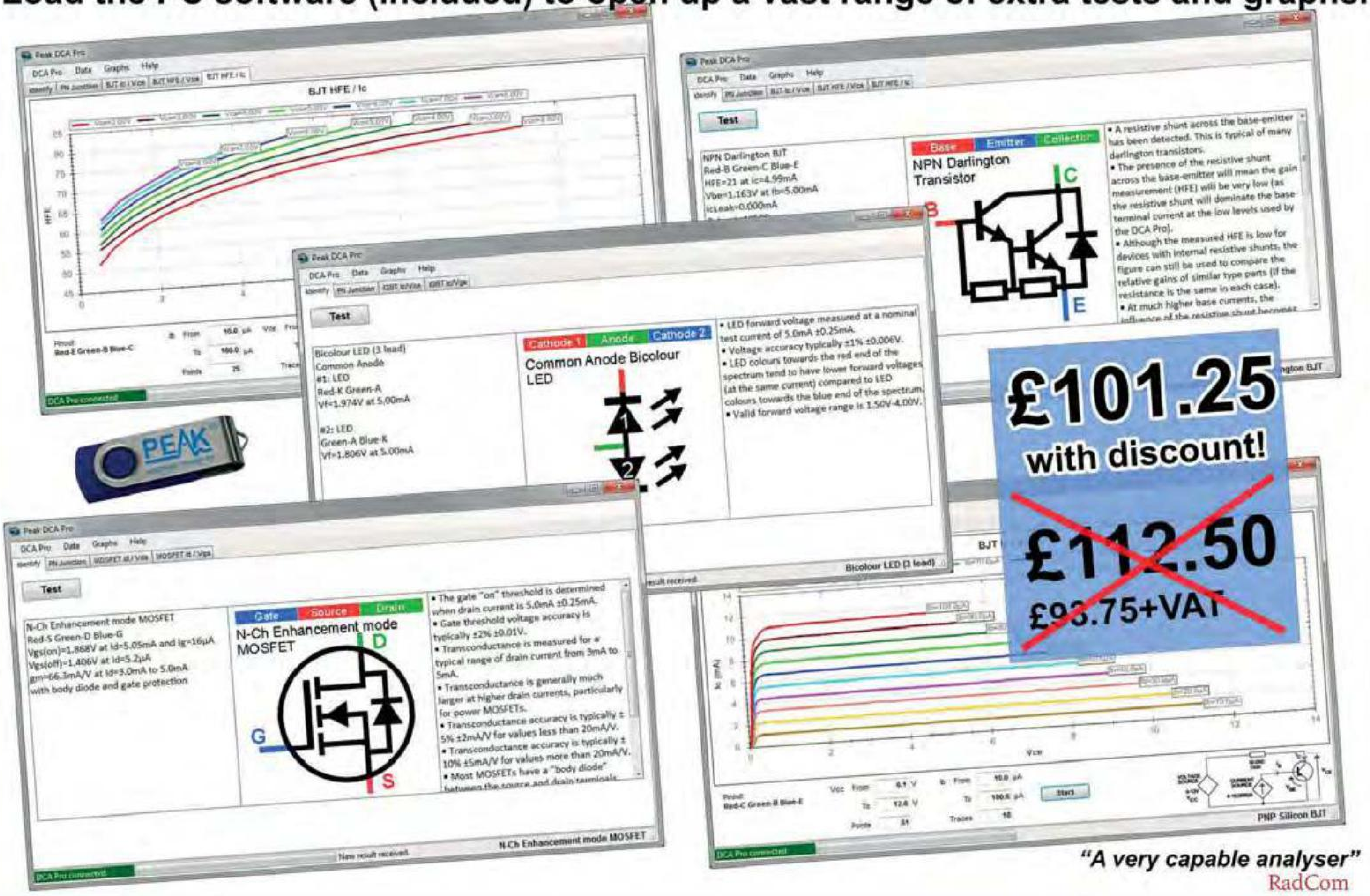
A versatile instrument that you'll love using for testing a huge range of semiconductor components.

Supports MOSFETs, transistors, diodes, LEDs, JFETs, IGBTs and more.

Connect any way round and let the ***DCA Pro*** identify the component type, pinout and lots of parameters.

Can be used standalone or connected to a Windows® PC (XP up to Win10).

Load the PC software (included) to open up a vast range of extra tests and graphs.


Peak DCA Pro
DCA Pro
Data
Graphs
Help
Identify
PN Junction
BJT HFE / Ic
Test
Base
Emitter
Collector
B
C
E
Cathode 1
Anode
Cathode 2
1
2
Gate
Source
Drain
D
G
S
PEAK
DCA Pro connected
New result received
PNP Silicon BJT

NPN Darlington BJT
Red-B Green-C Blue-E
HFE=21 at Ic=4.99mA
Vbe=1.163V at Ib=5.00mA
IcLeak=0.000mA

NPN Darlington Transistor

- A resistive shunt across the base-emitter has been detected. This is typical of many darlington transistors.
- The presence of the resistive shunt across the base-emitter will mean the gain measurement (HFE) will be very low (as the resistive shunt will dominate the base terminal current at the low levels used by the DCA Pro).
- Although the measured HFE is low for devices with internal resistive shunts, the figure can still be used to compare the relative gains of similar type parts (if the resistance is the same in each case).
- At much higher base currents, the

Bicolour LED (3 lead)
Common Anode
#1: LED
Red-K Green-A
Vf=1.974V at 5.00mA

#2: LED
Green-A Blue-K
Vf=1.806V at 5.00mA

Common Anode Bicolour LED

- LED forward voltage measured at a nominal test current of 5.0mA ±0.25mA.
- Voltage accuracy typically ±1% ±0.006V.
- LED colours towards the red end of the spectrum tend to have lower forward voltages (at the same current) compared to LED colours towards the blue end of the spectrum.
- Valid forward voltage range is 1.50V-4.00V.

N-Ch Enhancement mode MOSFET
Red-S Green-D Blue-G
Vgs(on)=1.868V at Id=5.05mA and Ig=16µA
Vgs(off)=1.406V at Id=5.2µA
gm=66.3mA/V at Id=3.0mA to 5.0mA
with body diode and gate protection

N-Ch Enhancement mode MOSFET

- The gate "on" threshold is determined when drain current is 5.0mA ±0.25mA.
- Gate threshold voltage accuracy is typically ±2% ±0.01V.
- Transconductance is measured for a typical range of drain current from 3mA to 5mA.
- Transconductance is generally much larger at higher drain currents, particularly for power MOSFETs.
- Transconductance accuracy is typically ± 5% ±2mA/V for values less than 20mA/V.
- Transconductance accuracy is typically ± 10% ±5mA/V for values more than 20mA/V.
- Most MOSFETs have a "body diode"

£101.25
with discount!

~~£112.50~~
~~£93.75+VAT~~

"A very capable analyser"
RadCom

It's only possible to show summary specifications here. Please ask if you'd like detailed data. Further information is also available on our website. Product price refunded if you're not happy.

Tel. 01298 70012
www.peakelec.co.uk
sales@peakelec.co.uk

Atlas House, 2 Kiln Lane
Harpur Hill Business Park
Buxton, Derbyshire
SK17 9JL, UK

UK designed **and** manufactured